Dell™ PowerConnect™ 6200 Series

Configuration Guide

Model: PC6224, PC6248, PC6224P, PC6248P, and PC6224F

Notes, Cautions, and Warnings

NOTE: A NOTE indicates important information that helps you make better use of your computer.

CAUTION: A CAUTION indicates potential damage to hardware or loss of data if instructions are not followed.

WARNING: A WARNING indicates a potential for property damage, personal injury, or death.

Information in this document is subject to change without notice.
© 2010 Dell Inc. All rights reserved.

Reproduction of these materials in any manner whatsoever without the written permission of Dell Inc. is strictly forbidden.

Trademarks used in this text: *Dell*, the *DELL* logo, and *PowerConnect* are trademarks of Dell Inc. *sFlow* is a registered trademark of InMon Corporation. *Cisco* is a registered trademark of Cisco Systems, Inc. and/or its affiliates in the United States and certain other countries.

Other trademarks and trade names may be used in this document to refer to either the entities claiming the marks and names or their products. Dell Inc. disclaims any proprietary interest in trademarks and trade names other than its own.

Model: PC6224, PC6248, PC6224P, PC6248P, and PC6224F

April 2010 **Rev. A04**

Contents

1 About this Document 9

Organization **9**

Additional Documentation **10**

2 System Configuration 11

Traceroute **12**

CLI Example 12

Configuration Scripting **13**

Overview 13

Considerations 13

CLI Examples 14

Outbound Telnet **16**

Overview 16

CLI Examples 17

Simple Network Time Protocol (SNTP) **17**

Overview 17

CLI Examples 18

Syslog **20**

Overview 20

CLI Examples 20

Port Description **22**

CLI Example 22

Storm Control **23**

CLI Example 23

Cable Diagnostics **25**

Copper Port Cable Test 25

Fiber Port Cable Test 27

3 Switching Configuration 29

Virtual LANs **29**
VLAN Configuration Example 30
CLI Examples 31
Web Interface 33
IP Subnet and MAC-Based VLANs 34
CLI Examples 34
Private Edge VLANs 35
CLI Example 36

Voice VLAN **37**
Using Voice VLAN 37
Interaction with LLDP-MED 38

IGMP Snooping **40**
CLI Examples 40

IGMP Snooping Querier **43**
CLI Examples 43

Link Aggregation/Port Channels **45**
CLI Example 46
Web Interface Configuration: LAGs/Port-channels 48

Port Mirroring **49**
Overview 49
CLI Examples 49

Port Security **50**
Overview 50
Operation 50
CLI Examples 51

Link Layer Discovery Protocol **52**
CLI Examples 52

Denial of Service Attack Protection **54**
Overview 54
CLI Examples 55

DHCP Snooping **56**
CLI Examples 59

sFlow **67**
Overview 67
sFlow Agents 68
CLI Examples 69

4 Routing Configuration 73

VLAN Routing **74**
CLI Examples 74
Using the Web Interface to Configure VLAN Routing 76

Virtual Router Redundancy Protocol **77**
CLI Examples 77
Using the Web Interface to Configure VRRP 79

Proxy Address Resolution Protocol (ARP) **80**
Overview 80
CLI Examples 80

OSPF **81**
OSPF Concepts and Terms 81
CLI Examples 83

Routing Information Protocol **92**
RIP Configuration 92
CLI Examples 93
Using the Web Interface to Configure RIP 94

Route Preferences **95**
Assigning Administrative Preferences to Routing Protocols 95
Using Equal Cost Multipath 97

Loopback Interfaces **99**

IP Helper **100**
CLI Examples 102

5 Device Security 105

802.1x Network Access Control **106**
802.1x Network Access Control Examples 106

802.1X Authentication and VLANs **109**
Authenticated and Unauthenticated VLANs 109
Guest VLAN 109
CLI Examples 110

Authentication Server Filter Assignment **111**

Access Control Lists (ACLs) **111**
Overview 111
MAC ACLs 113
IP ACLs 114
ACL Configuration Process 114
IP ACL CLI Example 115
Configuring a MAC ACL 116

RADIUS **117**
RADIUS Configuration Examples 118

TACACS+ **120**
TACACS+ Configuration Example 120

802.1x MAC Authentication Bypass (MAB) **122**
Operation in the Network 122
CLI Examples 123

Captive Portal **125**
Overview 125
Functional Description 125
Captive Portal Configuration, Status and Statistics 126
Captive Portal Status 128
Captive Portal Statistics 129
CLI Examples 129

6 IPv6 135

Overview **135**

Interface Configuration **135**

CLI Example 136

7 Quality of Service 139

Class of Service Queuing **139**

Ingress Port Configuration 139

Egress Port Configuration—Traffic Shaping 140

Queue configuration 140

Queue Management Type 140

CLI Examples 140

Differentiated Services **143**

CLI Example 144

DiffServ for VoIP Configuration Example 146

8 Multicast 149

Overview **149**

When to Enable IP Multicast on the PowerConnect 6200 Series Switch 150

IGMP Configuration **150**

CLI Example 150

IGMP Proxy **151**

CLI Examples 151

DVMRP **152**

CLI Example 153

PIM **154**

PIM-SM 154

PIM-DM 155

Multicast Routing and IGMP Snooping **157**

9 Utility 161

Auto Config **162**

Overview 162

Functional Description 162

CLI Examples 167

Nonstop Forwarding on a Switch Stack **168**

Initiating a Failover 168

Checkpointing 168

Switch Stack MAC Addressing and Stack Design Considerations . . . 170

NSF Network Design Considerations 170

NSF Default Behavior 170

Configuration Examples 171

1

About this Document

This configuration guide provides examples of how to use the Dell™ PowerConnect™ 6200 Series switch in a typical network. It describes the advantages of specific functions the PowerConnect 6200 Series switch provides and includes information about configuring those functions using the command line interface (CLI).

Organization

This document is organized as follows:

- "System Configuration" on page 11 describes how to configure basic system and port settings, use system interfaces and utilities, and create and use CLI scripts.
- "Switching Configuration" on page 29 provides configuration scenarios for layer 2 switching, including creating virtual local area networks (VLANs) and Internet Group Management Protocol (IGMP) snooping interfaces, and enabling port security.
- "Routing Configuration" on page 73 provides configuration scenarios for layer 3 features such as VLAN routing, Open Shortest Path First (OSPF), and Routing Information Protocol (RIP).
- "Device Security" on page 105 provides information on creating access control lists and configuring RADIUS and TACACS+ servers.
- "IPv6" on page 135 describes configuring and using IPv6-enabled interfaces in a mixed IPv6/IPv4 network.
- "Quality of Service" on page 139 provides configuration scenarios for class-of-service (CoS) queueing and differentiated services (DiffServ).
- "Multicast" on page 149 describes how to configure IGMP, IGMP proxy, Distance Vector Multicast Routing Protocol (DVMRP), and Protocol Independent Multicast (PIM) on the switch.
- "Utility" on page 161 describes the Auto Config and Nonstop Forwarding (NSF) features.

Additional Documentation

The following documentation provides additional information about PowerConnect 6200 Series software:

- The *CLI Command Reference* for your Dell PowerConnect switch describes the commands available from the command-line interface (CLI) for managing, monitoring, and configuring the switch.
- The *User's Guide* for your Dell PowerConnect switch describes the Web GUI. Many of the scenarios described in this document can be fully configured using the Web interface. This guide also provides initial system setup and configuration instructions.
- The *Getting Started Guide* for your Dell PowerConnect switch provides basic information to install, configure, and operate the system.
- Release notes for your Dell PowerConnect product detail the platform-specific functionality of the software packages, including issues and workarounds.

2

System Configuration

This section provides configuration scenarios for the following features:

- "Traceroute" on page 12
- "Configuration Scripting" on page 13
- "Outbound Telnet" on page 16
- "Simple Network Time Protocol (SNTP)" on page 17
- "Syslog" on page 20
- "Port Description" on page 22
- "Storm Control" on page 23
- "Cable Diagnostics" on page 25

NOTE: For information on setting up the hardware and serial or TFTP connection, refer to the *Getting Started Guide* for your system.

Traceroute

Use Traceroute to discover the routes that packets take when traveling on a hop-by-hop basis to their destination through the network.

- Maps network routes by sending packets with small Time-to-Live (TTL) values and watches the ICMP time-out announcements
- Command displays all L3 devices
- Can be used to detect issues on the network
- Tracks up to 30 hops
- Default UDP port uses 33434 unless modified in the traceroute command

CLI Example

The following shows an example of using the traceroute command to determine how many hops there are to the destination. The command output shows each IP address the packet passes through and how long it takes to get there. In this example, the packet takes 16 hops to reach its destination.

```
console#traceroute ?

ip                        Enter IP Address.
ipv6                      Use keyword 'ipv6' if entering IPv6 Address.

console#traceroute 72.14.253.99

Traceroute to 72.14.253.99 ,30 hops max 0 byte packets:

 1  10.131.10.1       <10 ms     <10 ms     <10 ms
 2  210.210.108.193   <10 ms      10 ms     <10 ms
 3  192.168.81.1      <10 ms      10 ms     <10 ms
 4  210.214.5.161     <10 ms      10 ms      10 ms
 5  210.214.5.169     <10 ms     <10 ms      10 ms
 6  124.7.202.2        10 ms     <10 ms     <10 ms
 7  210.18.7.166       40 ms      30 ms      30 ms
 8  202.144.2.193      30 ms      30 ms      30 ms
 9  202.144.113.151    30 ms      40 ms      30 ms
10  72.14.196.97       40 ms      30 ms     100 ms
11  216.239.43.216     40 ms      40 ms      30 ms
12  216.239.43.209     60 ms      40 ms      40 ms
13  216.239.43.222     40 ms      50 ms      50 ms
14  216.239.43.221    100 ms     110 ms     100 ms
15  209.85.250.88     130 ms     130 ms     120 ms
16  209.85.250.105    130 ms     120 ms     130 ms
17  209.85.250.91     160 ms     160 ms     160 ms
18  216.239.47.237    290 ms     240 ms     250 ms
19  216.239.46.211    240 ms     270 ms     250 ms
```

```
--More-- or (q)uit
20  64.233.174.99     250 ms     240 ms     250 ms

Hop Count = 20 Last TTL = 30 Test attempt = 90 Test Success = 90
```

Configuration Scripting

Configuration scripting allows you to generate a text-formatted script file that shows the current system configuration. You can generate multiple scripts and upload and apply them to more than one switch.

Overview

Configuration scripting:

- Provides scripts that can be uploaded from and downloaded to the system.
- Provides flexibility to create command configuration scripts.
- Can be applied to several switches.
- Can save up to ten scripts up to a maximum size of 2 MB of memory.
- Provides List, Delete, Apply, Upload, Download.
- Provides script format of one CLI command per line.

NOTE: The startup-config and backup-config scripts are not bound by the 2 MB memory limit.

Considerations

When you use configuration scripting, keep the following considerations in mind:

- The total number of scripts stored on the system is limited by NVRAM/FLASH size.
- The application of scripts is partial if the script fails. For example, if the script executes five of ten commands and the script fails, the script stops at five.
- Scripts cannot be modified or deleted while being applied.
- Validation of scripts checks for syntax errors only. It does not validate that the script will run.

CLI Examples

The following are examples of the commands used for configurations scripting.

Example #1: Viewing the Script Options

```
console#script ?

apply        Applies configuration script to the switch.
delete       Deletes a configuration script file from the switch.
list         Lists all configuration script files present on the switch.
show         Displays the contents of configuration script.
validate     Validate the commands of configuration script.
```

Example #2: Viewing and Deleting Existing Scripts

```
console#script list

Configuration Script Name        Size(Bytes)
-------------------------------- -----------
abc.scr                                  360
running-config                           360
startup-config                           796
test.scr                                 360

4 configuration script(s) found.
2046 Kbytes free.

console#script delete test.scr

Are you sure you want to delete the configuration script(s)? (y/n)y

1 configuration script(s) deleted.
```

Example #3: Applying a Script to the Active Configuration

```
console#script apply abc.scr

Are you sure you want to apply the configuration script? (y/n)y
.....
....
Configuration script 'abc.scr' applied.
```

Example #4: Copying the Active Configuration into a Script

Use this command to capture the running configuration into a script.

```
console#show running-config running-config.scr

Config script created successfully.
```

Example #5: Uploading a Configuration Script to the TFTP Server

Use this command to upload a configuration script to the TFTP server.

```
console#copy script abc.scr tftp://10.27.64.141/abc.scr

Mode........................................... TFTP
Set TFTP Server IP............................. 10.27.64.141
TFTP Path...................................... ./
TFTP Filename.................................. abc.scr
Data Type...................................... Config Script
Source Filename................................ abc.scr

Management access will be blocked for the duration of the transfer
Are you sure you want to start? (y/n) y

267 bytes transferred

File transfer operation completed successfully.
```

Example #6: Downloading a Configuration Script to the TFTP Server

Use this command to download a configuration script from the TFTP server to the switch.

```
console#copy tftp://10.27.64.141/abc.scr script abc.scr


Mode........................................... TFTP
Set TFTP Server IP............................. 10.27.64.141
TFTP Path...................................... ./
TFTP Filename.................................. abc.scr
Data Type...................................... Config Script
Destination Filename........................... abc.scr

Management access will be blocked for the duration of the transfer
Are you sure you want to start? (y/n) y

193 bytes transferred

Validating configuration script...
configure
```

```
exit
configure
logging web-session
bridge aging-time 100
exit

Configuration script validated.
File transfer operation completed successfully.
```

Example #7: Validating a Script

```
console#script validate abc.scr
ip address dhcp
username "admin" password 16d7a4fca7442dda3ad93c9a726597e4 level 15 encrypted
exit

Configuration script 'abc.scr' validated.

console#script apply abc.scr

Are you sure you want to apply the configuration script? (y/n)y

ip address dhcp
username "admin" password 16d7a4fca7442dda3ad93c9a726597e4 level 15 encrypted
exit

Configuration script 'abc.scr' applied.
```

Outbound Telnet

Overview

Outbound telnet:

- Establishes an outbound telnet connection between a device and a remote host.
- When a telnet connection is initiated, each side of the connection is assumed to originate and terminate at a "Network Virtual Terminal" (NVT).
- Server and user hosts do not maintain information about the characteristics of each other's terminals and terminal handling conventions.
- Must use a valid IP address.

CLI Examples

The following are examples of the commands used in the outbound telnet feature.

Example #1: Connecting to Another System by Using Telnet

```
console#telnet 192.168.77.151
Trying 192.168.77.151...
console#
User:admin
Password:
(Dell PC62XX Routing) >enable
Password:

console#show ip interface

Management Interface:

IP Address..................................... 10.27.65.89
Subnet Mask.................................... 255.255.254.0
Default Gateway................................ 10.27.64.1
Burned In MAC Address.......................... 00FF.F2A3.6688
Network Configuration Protocol Current......... DHCP
Management VLAN ID............................. 4086

Routing Interfaces:
                                                Netdir   Multi
Interface    IP Address      IP Mask            Bcast    CastFwd
----------   --------------- ---------------- -------- --------
```

Simple Network Time Protocol (SNTP)

Overview

The SNTP implementation has the following features:

- Used for synchronizing network resources
- Adaptation of NTP
- Provides synchronized network timestamp
- Can be used in broadcast or unicast mode
- SNTP client implemented over UDP that listens on port 123

CLI Examples

The following are examples of the commands used in the SNTP feature.

Example #1: Viewing SNTP Options

```
(Dell PC62XX Routing)(Config) #sntp ?

console(config)#sntp ?

authenticate            Require authentication for received Network Time
                        Protocol (NTP) traffic from servers.
authentication-key      Define an authentication key for Simple Network Time
                        Protocol (SNTP).
broadcast               Configure SNTP client broadcast parameters.
client                  Configure the SNTP client parameters.
server                  Configure SNTP server parameters.
trusted-key             Authenticate the identity of a system to which
                        SNTP will synchronize.
unicast                 Configure SNTP client unicast parameters.
```

Example #2: Configuring the SNTP Server

```
console(config)#sntp server ?

<ipaddress/domain-name>  Enter SNTP server address or the domain name.

console(config)#sntp server 192.168.10.25 ?

key                       Authentication key to use when sending packets to
                          this peer.
poll                      Enable/Disable SNTP server polling.
priority                  Configure SNTP server priority.
<cr>                      Press enter to execute the command.

console(config)#sntp server 192.168.10.25
```

Example #3: Viewing SNTP Information

```
console#show sntp ?

configuration            Show the configuration of the Simple Network Time
                         Protocol (SNTP).
status                   To show the status of the Simple Network Time
                         Protocol (SNTP).

console#show sntp configuration

Polling interval: 64 seconds
MD5 Authentication keys:
Authentication is not required for synchronization.
Trusted keys:
No trusted keys.
Unicast clients: Enable

Unicast servers:
Server          Key             Polling         Priority
---------       -----------     -----------     ----------
192.168.0.1     Disabled        Enabled         1

console#show sntp status

Unicast servers:
Server          Status          Last response
---------       -----------     -------------------------
192.168.10.25   Unknown         00:00:00 Jan 1 1970
```

Syslog

Overview

Syslog:

- Allows you to store system messages and/or errors.
- Can store to local files on the switch or a remote server running a syslog daemon.
- Provides a method of collecting message logs from many systems.

Interpreting Log Files

Figure 2-1 describes the information that displays in log messages.

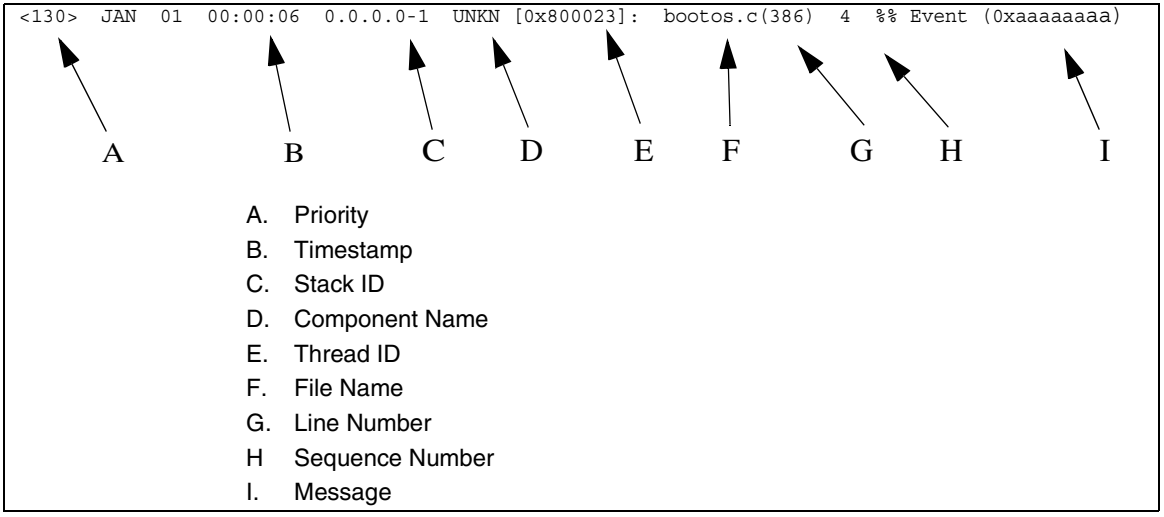


Figure 2-1. Log Files Key

CLI Examples

The following are examples of the commands used in the Syslog feature.

Example #1: Viewing Logging Information

```
console#show logging

Logging is enabled
Console Logging: level warning. Console Messages: 230 Dropped.
Buffer Logging: level info. Buffer Messages: 230 Logged.
File Logging: level notActive. File Messages: 0 Dropped.
CLI Command Logging : disabled
```

```
Web Session Logging : disabled
SNMP Set Command Logging : disabled
0 Messages were not logged.
Buffer Log:
<189> JAN 01 03:57:58 10.27.65.86-1 TRAPMGR[216282304]: traputil.c(908) 31 %%
Instance 0 has elected a new STP root: 8000:00ff:f2a3:8888
<189> JAN 01 03:57:58 10.27.65.86-1 TRAPMGR[216282304]: traputil.c(908) 32 %%
Instance 0 has elected a new STP root: 8000:0002:bc00:7e2c
<189> JAN 01 04:04:18 10.27.65.86-1 TRAPMGR[231781808]: traputil.c(908) 33 %% New
Spanning Tree Root: 0, Unit: 1
<189> JAN 01 04:04:18 10.27.65.86-1 TRAPMGR[216282304]: traputil.c(908) 34 %% The
unit 1 elected as the new STP root
```

Example #2: Viewing the Logging File

```
console#show logging file

Persistent Logging                  : disabled
Persistent Log Count                : 0
```

Example #5: Configuring Syslog Server

```
console(config)#logging ?

buffered                Buffered (In-Memory) Logging Configuration.
cli-command             CLI Command Logging Configuration.
console                 Console Logging Configuration.
facility                Syslog Facility Configuration.
file                    Configure logging file parameters.
on                      Enable logging to all supporting destinations.
snmp                    SNMP Set Command Logging Configuration.
web-session             Web Session Logging Configuration.
<ip-address|hostname>   Configure syslog server IP address or Hostname up to
                        63 characters in length


console(config)#logging 192.168.10.65

console(Config-logging)#?

description             Specify syslog server description.
exit                    To exit from the mode.
level                   Specify logging level.
port                    Specify UDP port (default is 514).

console(Config-logging)#level ?
```

```
alert                    Immediate action needed
critical                 Critical conditions
debug                    Debugging messages
emergency                System is unusable
error                    Error conditions
info                     Informational messages
notice                   Normal but significant conditions
warning                  Warning conditions

console(Config-logging)#level critical
```

Port Description

The Port Description feature lets you specify an alphanumeric interface identifier that can be used for SNMP network management.

CLI Example

Use the commands shown below for the Port Description feature.

Example #1: Enter a Description for a Port

This example specifies the name "Test" for port 1/g17:

```
console#configure
console(config)#interface ethernet 1/g17
console(config-if-1/g17)#description Test
console(config-if-1/g17)#exit
console(config)#exit
```

Example #2: Show the Port Description

```
console#show interfaces description ethernet 1/g17

Port  Description
----  -----------------------------------------------------------
1/g17 Test
```

Storm Control

A traffic storm occurs when incoming packets flood the LAN resulting in network performance degradation. The Storm Control feature protects against this condition.

The switch software provides broadcast, multicast, and unicast storm recovery for individual interfaces.

Unicast Storm Control protects against traffic whose MAC addresses are not known by the system.

For broadcast, multicast, and unicast storm control, if the rate of traffic ingressing on an interface increases beyond the configured threshold for that type, the traffic is dropped.

To configure storm control, you will enable the feature for all interfaces or for individual interfaces, and you will set the threshold (storm control level) beyond which the broadcast, multicast, or unicast traffic will be dropped.

Configuring a storm-control level also enables that form of storm-control. Disabling a storm-control level (using the "no" version of the command) sets the storm-control level back to default value and disables that form of storm-control. Using the "no" version of the "storm-control" command (not stating a "level") disables that form of storm-control but maintains the configured "level" (to be active next time that form of storm-control is enabled).

NOTE: The actual rate of ingress traffic required to activate storm-control is based on the size of incoming packets and the hard-coded average packet size of 512 bytes - used to calculate a packet-per-second (pps) rate - as the forwarding-plane requires pps versus an absolute rate Kbps. For example, if the configured limit is 10%, this is converted to ~25000 pps, and this pps limit is set in forwarding plane (hardware). You get the approximate desired output when 512bytes packets are used.

CLI Example

The following examples show how to configure the storm control feature an Ethernet interface. The interface number is 1/g17.

Example #1: Set Broadcast Storm Control for an Interface

```
console#configure

console(config)#interface ethernet 1/g17

console(config-if-1/g17)#storm-control broadcast ?

<cr>                    Press enter to execute the command.
level                   Configure storm-control thresholds.

console(config-if-1/g17)#storm-control broadcast level ?

<rate>          Enter the storm-control threshold as percent of port
                speed. Percent of port speed is converted to
                PacketsPerSecond based on 512 byte average packet
                size and applied to HW. Refer to documentation for
                further details.

console(config-if-1/g17)#storm-control broadcast level 7
```

Example #2: Set Multicast Storm Control for an Interface

```
console(config-if-1/g17)#storm-control multicast level 8
```

Example #3: Set Unicast Storm Control for an Interface

```
console(config-if-1/g17)#storm-control unicast level 5
```

Cable Diagnostics

This section describes:

- "Copper Port Cable Test" on page 25
- "Fiber Port Cable Test" on page 27

NOTE: Cable Diagnostics is supported on SFP/XFP ports but not on the Stacking/CX-4/SFP+/10GbaseT ports.

Copper Port Cable Test

The cable test feature enables you to determine the cable connection status on a selected port. The switch uses Time Domain Reflectometry (TDR) technology to determine the quality and characteristics of a copper cable attached to a port.

NOTE: The cable test feature is supported only for copper cable. it is not supported for optical fiber cable.

NOTE: The copper-related commands do not apply to the stacking, 10G BaseT, or CX-4 ports associated with these plug-in modules.

In privileged exec mode, enter `test copper-port tdr` *`unit/port`* to run the cable test on the specified port. One of the following statuses are returned:

- **Normal**: The cable is working correctly.
- **Open**: The cable is disconnected or there is a faulty connector.
- **Short**: There is an electrical short in the cable.
- **Cable Test Failed**: The cable status could not be determined. The cable may in fact be working.

The command also returns a cable length estimate if this feature is supported by the PHY for the current link speed. The length is displayed as the estimated length. Note that if the link is down and a cable is attached to a 10/100 Ethernet adapter, then the cable status may display as Open or Short because some Ethernet adapters leave unused wire pairs unterminated or grounded. Unknown is displayed if the cable length could not be determined.

If the port has an active link while the cable test is run, the link can go down for the duration of the test. The test may take several seconds to run.

To view cable status information for multiple ports, enter `show copper-ports tdr`. If the cable test has not been run on a port, the results indicate that the test has not been performed.

Example #1: Cable Test for Copper Ports

```
console#test copper-port tdr 1/g1

Cable Status..................................... Short
Cable Length..................................... 5m

console#show copper-ports tdr

Port      Result   Length [meters]   Date
-------   ------   ---------------   ---------------------
1/g1      Short    9                 Jan 01 1970 18:03:23
1/g2      Test has not been performed
1/g3      Test has not been performed
1/g4      Test has not been performed
1/g5      Test has not been performed
--More-- or (q)uit
```

NOTE: You can also run a cable test using the Web Interface. In the navigation tree, click **System > Diagnostics**.

Example #2: Show Copper Cable Length

Use the `show copper-ports cable-length` command in Privileged EXEC mode to display the estimated copper cable length attached to a port. The following example displays the estimated copper cable length attached to all ports.

```
console#show copper-ports cable-length

Port     Length [meters]

----     ---------------

1/g1     <50

1/g2     Copper not active

1/g3     110-140

1/g4     Fiber
```

Example #3: Show Last Time Domain Reflectometry Tests

Use the **show copper-ports tdr** command in Privileged EXEC mode to display the last Time Domain Reflectometry (TDR) tests on specified ports.

The following example displays the last TDR tests on all ports.

```
console#show copper-ports tdr

Port    Result     Length [meters]    Date

----    --------   ---------------    ---------------

1/g1    OK

1/g2    Short      50                 13:32:00 23 July 2004

1/g3    Test has not been preformed

1/g4    Open       128                13:32:08 23 July 2004

1/g5    Fiber      -                   -
```

Fiber Port Cable Test

Example #1: Show Optical Transceiver Diagnostics

Use the **show fiber-ports optical-transceiver** command in Privileged EXEC mode to display the optical transceiver diagnostics.

NOTE: The show fiber ports command is only applicable to the SFP combo ports and XFP ports (not the ports on the SFP+ plug-in module).

The following example displays the optical transceiver diagnostics.

```
console#show fiber-ports optical-transceiver

Port          Temp    Voltage   Current   Output    Input   TX      LOS

                                            Power     Power   Fault

-----------   ------  -------   -------   ------    -----   -----   ---

1/g3          w       OK        E         OK        OK      OK      OK

1/g4          OK      OK        OK        OK        OK      E       OK

1/g1          Copper

Temp - Internally measured transceiver temperature

Voltage - Internally measured supply voltage

Current - Measured TX bias current

Output Power - Measured TX output power in milliWatts

Input Power - Measured RX received power in milliWatts

TX Fault - Transmitter fault

LOS - Loss of signal
```

3

Switching Configuration

This section provides configuration scenarios for the following features:

- "Virtual LANs" on page 29
- "Voice VLAN" on page 37
- "IGMP Snooping" on page 40
- "IGMP Snooping Querier" on page 43
- "Link Aggregation/Port Channels" on page 45
- "Port Mirroring" on page 49
- "Port Security" on page 50
- "Link Layer Discovery Protocol" on page 52
- "Denial of Service Attack Protection" on page 54
- "DHCP Snooping" on page 56
- "sFlow" on page 67

Virtual LANs

Adding Virtual LAN (VLAN) support to a Layer 2 switch offers some of the benefits of both bridging and routing. Like a bridge, a VLAN switch forwards traffic based on the Layer 2 header, which is fast. Like a router, it partitions the network into logical segments, which provides better administration, security and management of multicast traffic.

A VLAN is a set of end stations and the switch ports that connect them. You can have many reasons for the logical division, for example, department or project membership. The only physical requirement is that the end station, and the port to which it is connected, both belong to the same VLAN.

Each VLAN in a network has an associated VLAN ID, which appears in the IEEE 802.1Q tag in the Layer 2 header of packets transmitted on a VLAN. An end station may omit the tag, or the VLAN portion of the tag, in which case the first switch port to receive the packet may either reject it or insert a tag using its default VLAN ID. A given port may handle traffic for more than one VLAN, but it can only support one default VLAN ID.

Two features let you define packet filters that the switch uses as the matching criteria to determine if a particular packet belongs to a particular VLAN.

- The IP-subnet Based VLAN feature lets you map IP addresses to VLANs by specifying a source IP address, network mask, and the desired VLAN ID.
- The MAC-based VLAN feature let packets originating from end stations become part of a VLAN according to source MAC address. To configure the feature, you specify a source MAC address and a VLAN ID.

The Private Edge VLAN feature lets you set protection between ports located on the switch. This means that a protected port cannot forward traffic to another protected port on the same switch.

The feature does not provide protection between ports located on different switches.

For information about authenticated, unauthenticated, and guest VLANs, see "802.1X Authentication and VLANs" on page 109.

VLAN Configuration Example

The diagram in this section shows a switch with four ports configured to handle the traffic for two VLANs. Port 1/g18 handles traffic for both VLANs, while port 1/g17 is a member of VLAN 2 only, and ports 1/g19 and 1/g20 are members of VLAN 3 only. The script following the diagram shows the commands you would use to configure the switch as shown in the diagram.

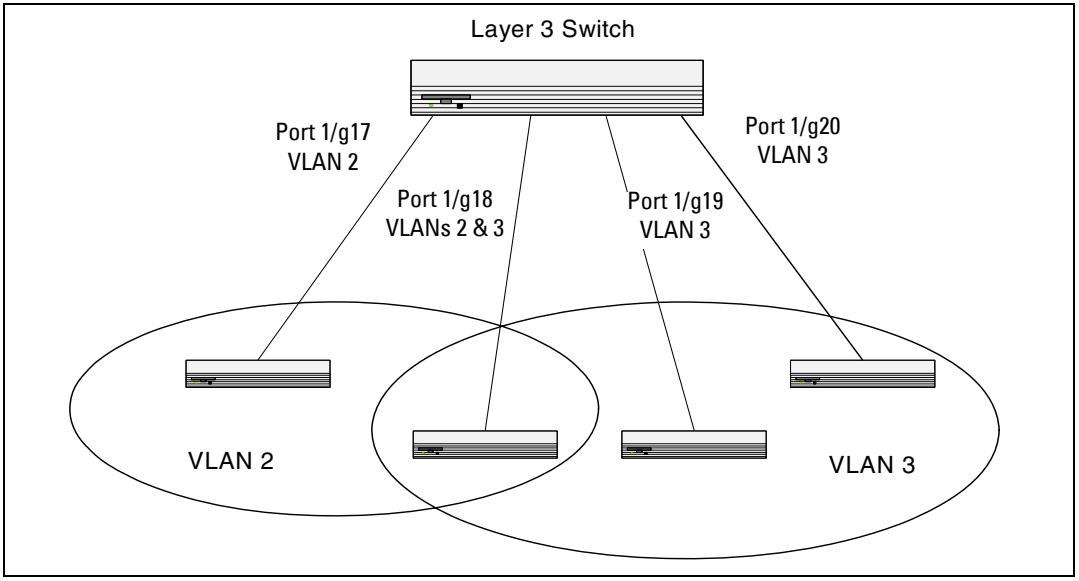


Figure 3-1. VLAN Example Network Diagram

CLI Examples

The following examples show how to create VLANs, assign ports to the VLANs, and assign a VLAN as the default VLAN to a port.

Example #1: Create Two VLANs

Use the following commands to create two VLANs and to assign the VLAN IDs while leaving the names blank.

```
console(config)#vlan database

console(config-vlan)#vlan 2

console(config-vlan)#vlan 3

console(config-vlan)#exit
```

Example #2: Assign Ports to VLAN2

This sequence shows how to assign ports to VLAN2, specify that frames will always be transmitted tagged from all member ports, and that untagged frames will be rejected on receipt.

```
console(config)#interface ethernet 1/g17

console(config-if-1/g17)#switchport mode general

console(config-if-1/g17)#switchport general allowed vlan add 2

console(config-if-1/g17)#switchport general acceptable-frame-type tagged-
only

console(config-if-1/g17)#exit

console(config)#interface ethernet 1/g18

console(config-if-1/g18)#switchport mode general

console(config-if-1/g18)#switchport general allowed vlan add 2

console(config-if-1/g18)#switchport general acceptable-frame-type tagged-
only

console(config-if-1/g18)#exit
```

Example #3: Assign Ports to VLAN3

This example shows how to assign the ports that will belong to VLAN 3. Untagged frames will be accepted on ports 1/g19 and 1/g20.

Note that port 1/g18 belongs to both VLANs and that port 1/g17 can never belong to VLAN 3.

```
console(config)#interface ethernet 1/g18

cconsole(config-if-1/g18)#switchport general allowed vlan add 3

console(config-if-1/g18)#exit

console(config)#interface ethernet 1/g19

console(config-if-1/g19)#switchport general allowed vlan add 3

console(config-if-1/g19)#exit

console(config)#interface ethernet 1/g20

console(config-if-1/g20)#switchport general allowed vlan add 3
```

Example #4: Assign VLAN3 as the Default VLAN

This example shows how to assign VLAN 3 as the default VLAN for port 1/g18.

```
console(config)#interface ethernet 1/g18

console(config-if-1/g18)#switchport general pvid 3
```

Example #5: Assign IP Addresses to VLAN 2

In order for the VLAN to function as a routing interface, you must enable routing on the VLAN and on the switch. Routing is only permitted on VLAN interfaces. Routing on physical interfaces is not supported.

```
console#configure

console(config)#interface vlan 2

console(config-if-vlan2)#ip address 192.168.10.33 255.255.255.0

console(config-if-vlan2)#routing

console(config-if-vlan2)#exit

console(config)#ip routing
```

Example #6: View Information About VLAN 2

```
console#show ip interface vlan 2


Primary IP Address.............................
192.168.10.33/255.255.255.0
Routing Mode.................................. Enable
Administrative Mode........................... Enable
Forward Net Directed Broadcasts............... Disable
Proxy ARP..................................... Enable
Local Proxy ARP............................... Disable
Active State.................................. Inactive
Link Speed Data Rate.......................... 10 Half
MAC Address................................... 00FF.F2A3.888A
Encapsulation Type............................ Ethernet
IP MTU........................................ 1500
```

Web Interface

Use the following screens to perform the same configuration using the Web Interface:

- **Switching > VLAN > Membership**. To create VLANs and specify port participation.
- **Switching > VLAN > Port Settings.** To specify the PVID and mode for the port.

IP Subnet and MAC-Based VLANs

In addition to port-based VLANs, the switch also supports VLANs that are based on the IP address or MAC address of a host. With IP subnet and MAC-based VLANs, the VLAN membership is determined by the address of the host rather than the port to which the host is attached.

CLI Examples

The following examples show how to associate an IP subnet with a VLAN, a specific IP address with a VLAN, and a MAC address with a VLAN.

Example #1: Associate an IP Subnet with a VLAN

This example shows how to configure the switch so that all hosts with IP addresses in the 192.168.25.0/24 network are members of VLAN 10.

```
console#configure

console(config)#vlan database

console(config-vlan)#vlan association subnet 192.168.25.0 255.255.255.0
10
```

Example #2: Associate an IP Address with a VLAN

This example shows how to configure the switch so a host with an IP addresses of 192.168.1.11 is a member of VLAN 10.

```
console#configure

console(config)#vlan database

console(config-vlan)#vlan association subnet 192.168.1.11
255.255.255.255 10
```

Example #3: Associate a MAC Address with a VLAN

This example shows how to configure the switch so a host with a MAC address of 00:ff:f2:a3:88:86 is a member of VLAN 10.

```
console#configure

console(config)#vlan database

console(config-vlan)#vlan association mac 00:ff:f2:a3:88:86 10
```

Example #4: Viewing IP Subnet and MAC-Based VLAN Associations

```
console#show vlan association mac


MAC Address         VLAN ID
-----------------   -------
00FF.F2A3.8886       10


console#show vlan association subnet


IP Subnet           IP Mask             VLAN ID
----------------    -----------------   -------
192.168.25.0                          255.255.255.0      10
192.168.1.11                          255.255.255.255    10
```

Private Edge VLANs

Use the Private Edge VLAN feature to prevent ports on the switch from forwarding traffic to each other even if they are on the same VLAN.

- Protected ports cannot forward traffic to other protected ports in the same group, even if they have the same VLAN membership. Protected ports can forward traffic to unprotected ports.
- Unprotected ports can forward traffic to both protected and unprotected ports.

You can also configure groups of protected ports, but unprotected ports are independent and cannot be added to a group. Each group's configuration consists of a name and a mask of ports. A port can belong to only one set of protected ports, but an unprotected port can be added to a group as a protected port.

The group name is configurable by the network administrator.

Use the `switchport protected` command to designate a port as protected. Use the `show switchport protected` command to display a listing of the protected ports.

CLI Example

Example #1: Configuring a Protected Port

The commands in this example name the protected port group 1 “PP_Test” and assign ports 1 and 2 to the group.

```
console(config)#switchport protected 1 name PP_Test


console(config)#interface ethernet 1/g17
console(config-if-1/g17)#switchport protected 1
console(config-if-1/g17)#exit


console(config)#interface ethernet 1/g18
console(config-if-1/g18)#switchport protected 1
console(config-if-1/g18)#exit
console(config)#exit
```

Example #2: Viewing Protected Port Group 1

```
console#show switchport protected 1



Name..................................... "PP_Test"

                                1/g17, 1/g18
```

Voice VLAN

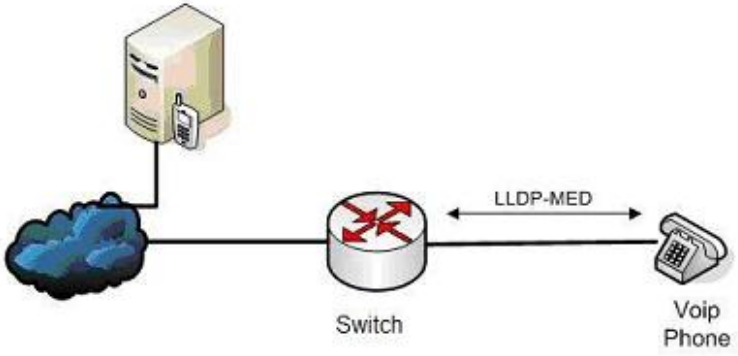
Voice VLAN enables switch ports to carry voice traffic with a defined priority in order to enable the separation of voice and data traffic coming onto the port. A primary benefit of using Voice VLAN is to ensure that the sound quality of an IP phone is safeguarded from deteriorating when the data traffic on the port is high.

The inherent isolation provided by VLANs ensures that inter-VLAN traffic is under management control and that network attached clients cannot initiate a direct attack on voice components. QoS based on IEEE 802.1P class of service (CoS) uses classification and scheduling to send network traffic from the switch in a predictable manner. The system uses the source MAC address of the traffic traveling through the port to identify the IP phone data flow.

IP Phones will use this VLAN. They will obtain their VLAN ID via CDP, DHCP or LLDP-MED. The voice traffic is sent to the switch tagged. The setup protocols (CDP, DHCP, etc.) are not tagged.

Using Voice VLAN

When an IP phone is connected to the switch, the voice traffic from the phone and the data traffic from the network could potentially deteriorate the voice quality. You can overcome this in multiple ways using different options in Voice VLAN.



You can configure the switch to support Voice VLAN on a port that is connecting the VOIP phone. Both of the following methods segregate the voice traffic and the data traffic in order to provide better service to the voice traffic.

- When a VLAN is associated with the Voice VLAN port, then the VLAN ID information is passed onto the VOIP phone using the LLDP-MED mechanism. By this method, the voice data coming from the VOIP phone is tagged with the exchanged VLAN ID, thus regular data arriving on the switch is given the default PVID of the port, and the voice traffic is received on a pre-defined VLAN. As a result, both kinds of traffic are segregated in order to provide better service to the voice traffic.

- When a dot1p priority is associated with the Voice VLAN port instead of a VLAN ID, then the priority information is passed onto the VOIP phone using the LLDP-MED mechanism. By this method, the voice data coming from the VOIP phone is tagged with VLAN 0 and with the exchanged priority; thus regular data arriving on the switch is given the default priority of the port (default 0), and the voice traffic is received with a higher priority.

You can configure the switch to override the data traffic CoS. This feature can override the 802.1 priority of the data traffic packets arriving at the port enabled for Voice VLAN. Therefore, any rogue client that is also connected to the Voice VLAN port does not deteriorate the voice traffic.

Interaction with LLDP-MED

The interactions with LLDP-MED are important for Voice VLAN:

- LLDP-MED notifies the Voice VLAN component of the presence and absence of a VoIP phone on the network.
- The Voice VLAN component interacts with LLDP-MED for applying VLAN ID, priority and tag information to the VoIP phone traffic.

For release 2.0 and earlier, the Voice VLAN feature can only be used by IP phones that support LLDP-MED, e.g. 4610SW Avaya phones.

Example#1: Configuring Voice VLAN

The commands in this example create a VLAN for voice traffic with a VLAN ID of 25. Then, Voice VLAN is administratively enabled on the switch. Finally, port 1/g12 is set to an 802.1Q VLAN and then enabled for Voice VLAN traffic.

```
console#configure
console(config)#vlan database
console(config-vlan)#vlan 25
console(config-vlan)#exit

console(config)#voice vlan

console(config)#interface ethernet 1/g12
console(config-if-1/g12)#switchport mode general
console(config-if-1/g12)#voice vlan 25
console(config-if-1/g12)#exit
console(config)#exit

console#show voice vlan interface 1/g12

Interface..................................... 1/g12
Voice VLAN Interface Mode..................... Enabled
Voice VLAN ID................................. 25
Voice VLAN COS Override....................... False
Voice VLAN Port Status........................ Disabled
Voice VLAN Authentication..................... Enabled
```

Example #2: Configuring Voice VLAN on an Unauthenticated Port

In some networks, multiple devices (for example, a PC, Printer, and phone) are connected to a single port on the switch. The PCs and printers are authenticated by 802.1X, but the phone might not support 802.1X authentication. The PowerConnect 6200 Series switches can allow unauthenticated traffic on the Voice VLAN for the phones that do not support authentication while requiring all other devices on the port to authenticate individually.

The phones that do not support 802.1X authentication are automatically directed to the Voice VLAN without manual configuration. The phones will obtain this information using one of the following methods:

- LLDP-MED
- CDP
- DHCP

In this example, interface 1/g10 is set to an 802.1Q VLAN. The port must be in general mode in order to enable MAC-based 802.1X authentication. Then, port 1/g10 is configured with MAC-based port authentication to allow authentication for multiple hosts on the same port (see "Example #2: MAC-Based Authentication Mode" on page 108 for more information). Next, Voice VLAN is enabled on the port with the Voice VLAN ID set to 25. Finally, Voice VLAN authentication is disabled on port 1/g10 because the phone connected to that port does not support 802.1X authentication. All other devices are required to use 802.1X authentication for network access.

Support for unauthenticated Voice VLANs is available in release 2.1 and later versions.

```
console#configure
console(config)#interface ethernet 1/g10

console(config-if-1/g10)#switchport mode general
console(config-if-1/g10)#dot1x port-control mac-based
console(config-if-1/g10)#voice vlan 25
console(config-if-1/g10)#voice vlan auth disable
console(config-if-1/g10)#<CTRL+Z>
console#show voice vlan interface 1/g10

Interface..................................... 1/g10
Voice VLAN Interface Mode..................... Enabled
Voice VLAN ID................................. 25
Voice VLAN COS Override....................... False
Voice VLAN Port Status........................ Disabled
Voice VLAN Authentication..................... Disabled
```

IGMP Snooping

This section describes the Internet Group Management Protocol (IGMP) Snooping feature. IGMP Snooping enables the switch to monitor IGMP transactions between hosts and routers. It can help conserve bandwidth by allowing the switch to forward IP multicast traffic only to connected hosts that request multicast traffic.

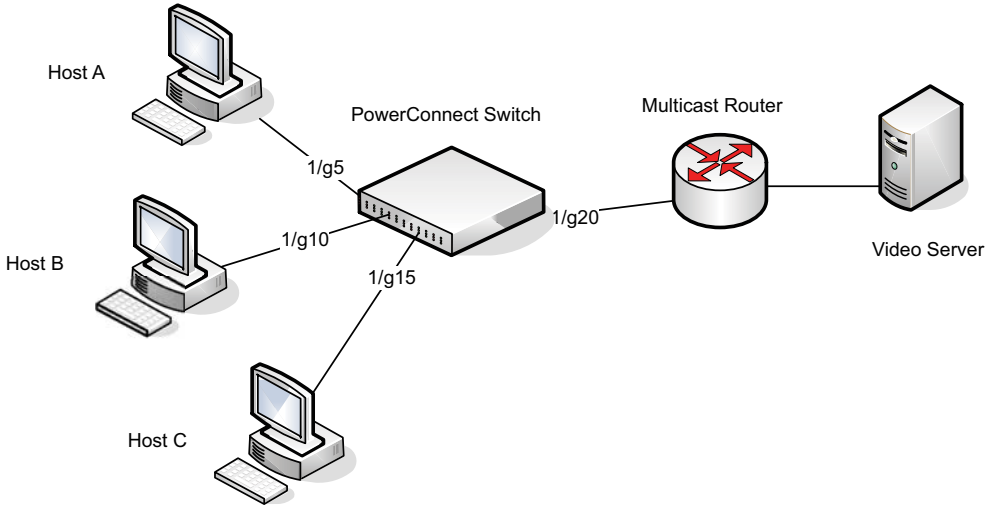
If you enable IGMP Snooping on the switch to listen to IGMP traffic, you do not need to enable IGMP, a layer 3 multicast protocol. IGMP Snooping is a layer 2 feature that allows the switch to dynamically add or remove ports from IP multicast groups by listening to IGMP join and leave requests. If you use the switch as a multicast router that can route multicast traffic between VLAN routing interfaces, you must enable a multicast routing protocol on the switch, such as PIM-SM. In this case, you can enable both IGMP and IGMP Snooping so that the switch routes IGMP traffic between VLANs and examines the IGMP packets for join and leave information. For information about configuring the PowerConnect 6200 Series switch as a mutlicast router that also performs IGMP snooping, see "Multicast Routing and IGMP Snooping" on page 157.

IGMP snooping can be enabled per VLAN. The IGMP feature on the PowerConnect 6200 Series switches uses IGMPv3 by default.

CLI Examples

In this example, the PowerConnect 6200 Series switch is a L2 switch with one non-default VLAN, VLAN 100. The three hosts are connected to ports that are members of VLAN 100, and IGMP snooping is enabled on VLAN 100. Port 1/g20 connects the switch to the L3 multicast router and is also a member of VLAN 100.

Figure 3-2. Switch with IGMP Snooping



1. Create VLAN 100.

```
console#configure
console(config)#vlan database
console(config-vlan)#vlan  100
```

2. Enable IGMP snooping on the VLAN.

```
console(config-vlan)#ip igmp snooping 100
console(config-vlan)#exit
```

3. Forbid the forwarding of unregistered multicast addresses on VLAN 100 to prevent multicast flooding to ports if there are no "listeners."

```
console(config)#interface vlan 100
console(config-if-vlan100)#bridge multicast forbidden
forward-unregistered
console(config-if-vlan100)#exit
```

4. Globally enable IGMP on the switch.

```
console(config)#ip igmp snooping
```

5. Configure port 1/g5 as a member of VLAN 100.

```
console(config)#interface ethernet 1/g5
console(config-if-1/g5)#switchport access vlan 100
console(config-if-1/g5)#exit
```

6. Configure port 1/g10 as a member of VLAN 100.

```
console(config)#interface ethernet 1/g10
console(config-if-1/g10)#switchport access vlan 100
console(config-if-1/g10)#exit
```

7. Configure port 1/g15 as a member of VLAN 100.

```
console(config)#interface ethernet 1/g15
console(config-if-1/g15)#switchport access vlan 100
console(config-if-1/g15)#exit
```

8. Configure port 1/g20 as a member of VLAN 100.

```
console(config)#interface ethernet 1/g20
console(config-if-1/g20)#switchport access vlan 100
console(config-if-1/g20)#exit
```

9. View information about the IGMP snooping configuration.

```
console#show ip igmp snooping

Admin Mode..................................... Enable

Multicast Control Frame Count.................. 0

Interfaces Enabled for IGMP Snooping........... None

Vlans enabled for IGMP snooping................ 100
```

In this example, Host A sends a join message for group 225.1.1.1. Host B sends a join message for group 225.1.1.2. Because IGMP snooping is enabled on the switch and on VLAN 100, the switch listens to the messages and dynamically adds ports 1/g5 and 1/g10 to the multicast address table. Port 1/g15 did not send a join message, so it does not appear in the table, as the following `show` command indicates.

```
console#show bridge multicast address-table


Vlan        MAC Address         Type      Ports

----  ----------------------  -------  -----------------

100   0100.5E01.0101          Dynamic    1/g5

100   0100.5E01.0102          Dynamic    1/g10

Forbidden ports for multicast addresses:

Vlan        MAC Address             Ports

----  ----------------------  ----------------------

100   0100.5E01.0101

100   0100.5E01.0102
```

When the video server sends multicast data to group 225.1.1.1, port 1/g5 participates and receives multicast traffic, but port 1/g10 does not participate because it is a member of a different multicast group. Without IGMP snooping, all ports that are members of VLAN 100 would be flooded with traffic for all multicast groups, which would greatly increase the amount of traffic on the switch.

You can use the show statistics command to view information about the multicast data transmitted or received by each interface. The following output shows a portion of the command output for interfaces 1/g5 and 1/g10. In this example, the counters were cleared before the video server began transmitting data.

```
console#show statistics ethernet 1/g5

...

Total Packets Received Without Errors.......... 626494

Unicast Packets Received....................... 0
```

```
Multicast Packets Received..................... 626494
Broadcast Packets Received..................... 0
console#show statistics ethernet 1/g10
...
Total Packets Received Without Errors.......... 12
Unicast Packets Received....................... 0
Multicast Packets Received..................... 12
Broadcast Packets Received..................... 0
```

IGMP Snooping Querier

When PIM and IGMP are enabled in a network with IP multicast routing, the IP multicast router acts as the IGMP querier. However, if the IP-multicast traffic in a VLAN needs to be Layer 2 switched only, an IP-multicast router is not required. The IGMP Snooping Querier can perform the IGMP snooping functions on the VLAN.

NOTE: Without an IP-multicast router on a VLAN, you must configure another switch as the IGMP querier so that it can send queries.

When the IGMP snooping querier is enabled, the IGMP snooping querier sends out periodic IGMP queries that trigger IGMP report messages from the switch that wants to receive IP multicast traffic. The IGMP snooping feature listens to these IGMP reports to establish appropriate forwarding.

CLI Examples

The following examples show commands to use with the IGMP Snooping Querier feature.

Example #1: Enable IGMP Snooping Querier on the Switch

The first command in this example enables the IGMP snooping querier on the switch. The second command specifies the IP address that the snooping querier switch should use as the source address when generating periodic queries.

```
console(config)#ip igmp snooping
console(config)#ip igmp snooping querier
console(config)#ip igmp snooping querier address 10.10.20.12
```

NOTE: The IGMP snooping feature must be enabled for the IGMP snooping querier function to operate.

Example #2: Configure IGMP Snooping Querier Properties

The first command in this example sets the IGMP Querier Query Interval time to 100. This means that the switch waits 100 seconds before sending another general query. The second command sets the IGMP Querier timer expiration period to 100. This means that the switch remains in Non-Querier mode for 100 seconds after it has discovered that there is a Multicast Querier in the network.

```
console(config)#ip igmp snooping querier query-interval 100

console(config)#ip igmp snooping querier timer expiry 100
```

Example #3: Show IGMP Snooping Querier Information

```
console#show ip igmp snooping querier


Global IGMP Snooping querier status

-----------------------------------

IGMP Snooping Querier Mode..................... Enable

Querier Address................................ 10.10.10.33

IGMP Version................................... 2

Querier Query Interval......................... 100

Querier Expiry Interval........................ 100
```

Example #4: Enable IGMP Snooping Querier on a VLAN

To configure IGMP Snooping Querier on a VLAN, enter VLAN Database mode. The first `ip igmp snooping` command in this example enables the IGMP snooping querier on VLAN 10. The second `ip igmp snooping` command specifies the IP address that the snooping querier switch should use as source address when generating periodic queries. The final command enables the Snooping Querier to participate in the Querier Election process when it discovers the presence of another Querier in the VLAN.

NOTE: For IGMP Snooping Querier functionality to be operationally enabled on the VLAN, IGMP Snooping and IGMP Snooping Querier must both be enabled globally on the switch.

```
console(config)#vlan database

console(config-vlan)#ip igmp snooping querier 10

console(config-vlan)#ip igmp snooping querier 10 address 10.10.11.40

console(config-vlan)#ip igmp snooping querier election participate 10
```

Example #5: Show IGMP Snooping Querier Information for VLAN 10

```
console#show ip igmp snooping querier vlan 10


Vlan 10  :   IGMP Snooping querier status
-------------------------------------------
IGMP Snooping Querier Vlan Mode.............. Enable
Querier Election Participate Mode.............. Enable
Querier Vlan Address.......................... 10.10.11.40
Operational State............................. Querier
Operational version........................... 2
Operational Max Resp Time..................... 10
```

Link Aggregation/Port Channels

This section shows how to use the Link Aggregation feature to configure port-channels via the Command Line Interface and the Graphical User Interface.

The Link Aggregation (LAG) feature allows the switch to treat multiple physical links between two end-points as a single logical link called a port-channel. All of the physical links in a given port-channel must operate in full-duplex mode at the same speed.

You can use the feature to directly connect two switches when the traffic between them requires high bandwidth and reliability, or to provide a higher bandwidth connection to a public network.

You can configure the port-channels as either dynamic or static. Dynamic configuration uses the IEEE 802.3ad standard, which provides for the periodic exchanges of LACPDUs. Static configuration is used when connecting the switch to an external switch that does not support the exchange of LACPDUs.

The feature offers the following benefits:

- Increased reliability and availability: If one of the physical links in the port-channel goes down, traffic is dynamically and transparently reassigned to one of the other physical links.
- Increased bandwidth: The aggregated physical links deliver higher bandwidth than each individual link.
- Incremental increase in bandwidth: A physical upgrade could produce a 10-times increase in bandwidth; LAG produces a two- or five-times increase, useful if only a small increase is needed.

Management functions treat a port-channel as if it were a single physical port.

You can include a port-channel in a VLAN. You can configure more than one port-channel for a given switch.

CLI Example

The following shows an example of configuring the software to support Link Aggregation (LAG) to a server and to a Layer 3 switch.

Figure 3-3 shows the example network.

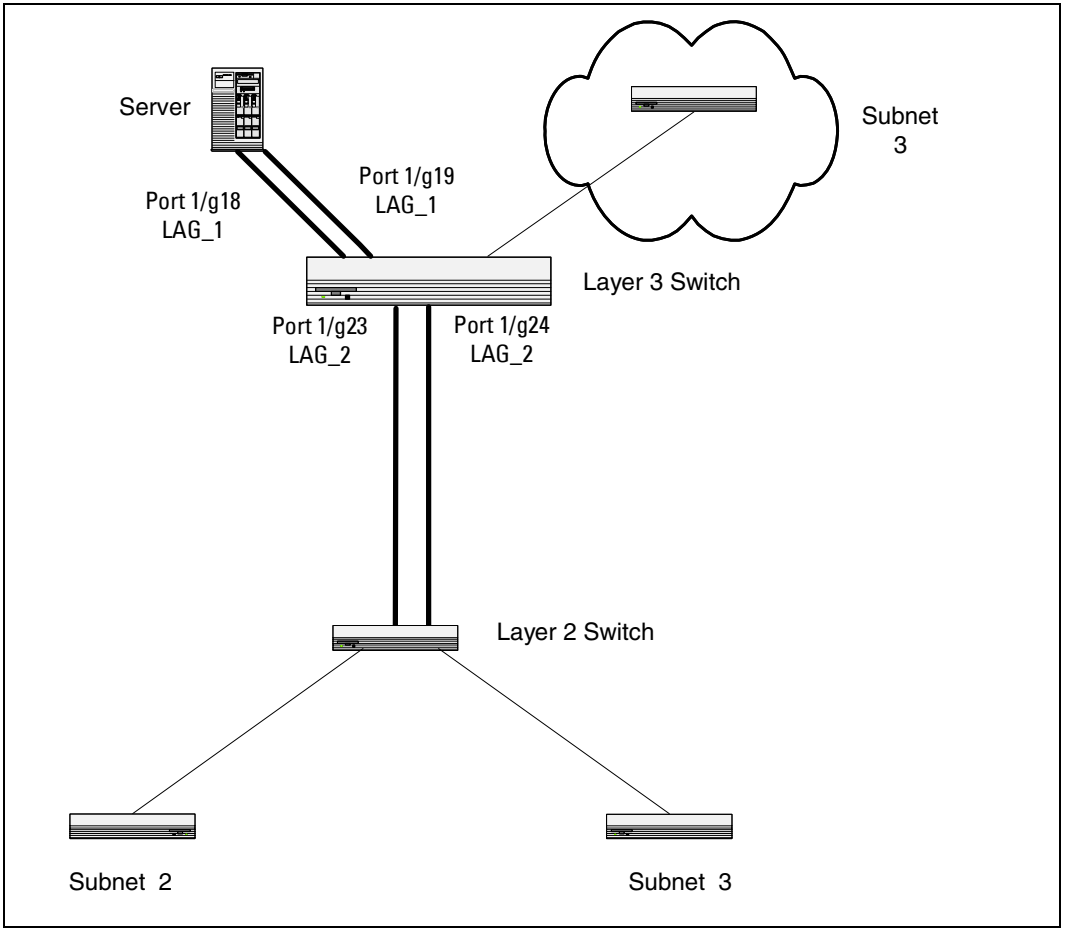


Figure 3-3. LAG/Port-channel Example Network Diagram

Example 1: Create Names for Two Port-Channels

```
console#configure
console(config)#interface port-channel 1
console(config-if-ch1)#description lag_1
console(config-if-ch1)#exit
console(config)#interface port-channel 2
console(config-if-ch2)#description lag_2
console(config-if-ch2)#exit
```

Example 2: Add the Physical Ports to the Port-Channels

```
console(config)#interface ethernet 1/g18
console(config-if-1/g18)#channel-group 1 mode auto
console(config-if-1/g18)#exit
console(config)#interface ethernet 1/g19
console(config-if-1/g19)#channel-group 1 mode auto
console(config-if-1/g19)#exit
console(config)#interface ethernet 1/g23
console(config-if-1/g23)#channel-group 2 mode auto
console(config-if-1/g238)#exit
console(config)#interface ethernet 1/g24
console(config-if-1/g24)#channel-group 2 mode auto
console(config-if-1/g24)#exit
console(config)#exit
```

Example 3: Show the Port Channels

By default, the system enables link trap notification

```
console#show interfaces port-channel

Channel    Ports                          Hash Algorithm Type
-------    ------------------------------ ------------------
ch1        No Configured Ports            3
```

```
ch2       No Configured Ports          3
ch3       No Configured Ports          3
ch4       No Configured Ports          3
ch5       No Configured Ports          3
ch6       No Configured Ports          3
ch7       No Configured Ports          3
ch8       No Configured Ports          3
ch9       No Configured Ports          3
ch10      No Configured Ports          3
ch11      No Configured Ports          3
ch12      No Configured Ports          3
ch13      No Configured Ports          3
ch14      No Configured Ports          3
ch15      No Configured Ports          3
ch16      No Configured Ports          3
ch17      No Configured Ports          3
ch18      No Configured Ports          3
ch19      No Configured Ports          3
ch20      No Configured Ports          3
```

At this point, the LAGs could be added to the default management VLAN.

Web Interface Configuration: LAGs/Port-channels

To perform the same configuration using the Graphical User Interface, click **Switching > Link Aggregation > LAG Membership** in the navigation tree.

Port Mirroring

This section describes the Port Mirroring feature, which can serve as a diagnostic tool, debugging tool, or means of fending off attacks.

Overview

Port mirroring selects network traffic from specific ports for analysis by a network analyzer, while allowing the same traffic to be switched to its destination. You can configure many switch ports as source ports and one switch port as a destination port. You can also configure how traffic is mirrored on a source port. Packets received on the source port, transmitted on a port, or both received and transmitted, can be mirrored to the destination port.

CLI Examples

The following are examples of the commands used in the Port Mirroring feature.

Example #1: Set up a Port Mirroring Session

The following command sequence enables port mirroring and specifies a source and destination ports.

```
console#configure
console(config)#monitor session 1 mode
console(config)#monitor session 1 source interface 1/g7 ?

rx                          Monitor ingress packets only.
tx                          Monitor egress packets only.
<cr>                        Press enter to execute the command.

console(config)#monitor session 1 source interface 1/g7
console(config)#monitor session 1 destination interface 1/g10
console(config)#exit
```

Example #2: Show the Port Mirroring Session

```
console#show monitor session 1

Session ID   Admin Mode   Probe Port   Mirrored Port   Type
----------   ----------   ----------   -------------   -----
1            Enable       1/g10        1/g7            Rx,Tx
```

Port Security

This section describes the Port Security feature.

Overview

Port Security:

- Allows for limiting the number of MAC addresses on a given port.
- Packets that have a matching MAC address (secure packets) are forwarded; all other packets (unsecure packets) are restricted.
- Enabled on a per port basis.
- When locked, only packets with allowable MAC address will be forwarded.
- Supports both dynamic and static.
- Implement two traffic filtering methods. These methods can be used concurrently.
  - Dynamic Locking: User specifies the maximum number of MAC addresses that can be learned on a port. The maximum number of MAC addresses is 100. After the limit is reached, additional MAC addresses are not learned. Only frames with an allowable source MAC address are forwarded.
  - Static Locking: User manually specifies a list of static MAC addresses for a port.

Operation

Port Security:

- Helps secure network by preventing unknown devices from forwarding packets.
- When link goes down, all dynamically locked addresses are 'freed.'
- If a specific MAC address is to be set for a port, set the dynamic entries to 0, then only allow packets with a MAC address matching the MAC address in the static list.
- Dynamically locked MAC addresses are aged out if another packet with that address is not seen within the age-out time. The user can set the time-out value.
- Dynamically locked MAC addresses are eligible to be learned by another port.
- Static MAC addresses are not eligible for aging.

CLI Examples

The following are examples of the commands used in the Port Security feature.

Example #1: Enable Port Security on an Interface

```
console(config)#interface ethernet 1/g18
console(config-if-1/g18)#port security ?

<cr>                        Press enter to execute the command.
discard                     Discard frames with unlearned source addresses.
max                         Configure the maximum addresses that can be
learned
                            on the port.
trap                       Sends SNMP Traps, and specifies the minimum time
                            between consecutive traps.

console(config-if-1/g18)#port security
```

Example #2: Show Port Security

```
console#show ports security ?

addresses                   Addresses.
ethernet                    Ethernet port.
port-channel                Link Aggregation interface.
<cr>                        Press enter to execute the command.
Example #3: Show Port Security on an Interface
console#show ports security ethernet 1/g18

Port    Status     Action              Maximum  Trap      Frequency
-----   --------   -----------------   -------  -------   ---------

1/g18    Locked     Discard             100       Disable  30
```

Link Layer Discovery Protocol

The Link Layer Discovery Protocol (LLDP) feature allows individual interfaces on the switch to advertise major capabilities and physical descriptions. Network managers can view this information and identify system topology and detect bad configurations on the LAN.

LLDP has separately configurable transmit and receive functions. Interfaces can transmit and receive LLDP information.

CLI Examples

Example #1: Set Global LLDP Parameters

Use the following sequence to specify switch-wide notification interval and timers for all LLDP interfaces.

```
console(config)#lldp ?


notification-interval                    Configure minimum interval to
send remote data

                                         change notifications.
timers                                   Configure the LLDP global
timer values.
console(config)#lldp notification-interval 1000
console(config)#lldp timers hold 8 reinit 5
console(config)#exit
```

Example #2: Set Interface LLDP Parameters

The following commands configure the Ethernet interface 1/g10 to transmit and receive LLDP information.

```
console#configure
console(config)#interface ethernet 1/g10
console(config-if-1/g10)#lldp receive
console(config-if-1/g10)#lldp transmit
console(config-if-1/g10)#lldp transmit-mgmt
console(config-if-1/g10)#exit
console(config)#exit
```

Example #3: Show Global LLDP Parameters

```
console#show lldp

LLDP Global Configuration

Transmit Interval.......................... 30 seconds
Transmit Hold Multiplier................... 8
Reinit Delay............................... 5 seconds
Notification Interval...................... 1000 seconds
```

Example #4 Show Interface LLDP Parameters

```
console#show lldp interface 1/g10

LLDP Interface Configuration

Interface  Link    Transmit  Receive   Notify    TLVs     Mgmt
---------  ------  --------  --------  --------  -------  ----
1/g10      Down    Enabled   Enabled   Disabled           Y

TLV Codes: 0- Port Description,   1- System Name
           2- System Description, 3- System Capabilities
```

Denial of Service Attack Protection

This section describes the PowerConnect 6200 Series Denial of Service Protection feature.

Overview

Denial of Service:

- Spans two categories:
  - Protection of the switch
  - Protection of the network
- Protects against the exploitation of a number of vulnerabilities which would make the host or network unstable
- Compliant with Nessus. Dell tested the switch software with Nessus version 2.0.10. Nessus is a widely-used vulnerability assessment tool.
- PowerConnect 6200 Series software provides a number of features that help a network administrator protect networks against DoS attacks.

There are 6 available types of attacks which can be monitored for and blocked. Each type of attack is represented by a `dos-control` command keyword.

```
console(config)#dos-control ?


firstfrag                Enables IPv4 first fragment checking.
icmp                     Enables ICMP size checking.
l4port                   Enables L4 port number checking.
sipdip                   Enables SIP=DIP checking.
tcpflag                  Enables TCP flag checking.
tcpfrag                  Enables TCP fragment checking.
```

Table 3-1 describes the `dos-control` keywords.

Table 3-1. DoS Control

Keyword	Meaning
firstfrag	Enabling First Fragment DoS prevention causes the switch to drop packets that have a TCP header smaller then the configured Min TCP Hdr Size.
icmp	ICMP DoS prevention causes the switch to drop ICMP packets that have a type set to ECHO_REQ (ping) and a size greater than the configured ICMP Pkt Size.
l4port	Enabling L4 Port DoS prevention causes the switch to drop packets that have TCP/UDP source port equal to TCP/UDP destination port.
sipdip	Enabling SIP=DIP DoS prevention causes the switch to drop packets that have a source IP address equal to the destination IP address.
tcpflag	Enabling TCP Flag DoS prevention causes the switch to drop packets that have TCP flag SYN set and TCP source port less than 1024 or TCP control flags set to 0 and TCP sequence number set to 0 or TCP flags FIN, URG, and PSH set and TCP sequence number set to 0 or both TCP flags SYN and FIN set.
tcpfrag	Enabling TCP Fragment DoS prevention causes the switch to drop packets that have an IP fragment offset equal to 1.

CLI Examples

The commands shown below show how to enable DoS protection and view its status.

Example #1: Enabling all DOS Controls

```
console#configure
console(config)#dos-control sipdip
console(config)#dos-control firstfrag
console(config)#dos-control tcpfrag
console(config)#dos-control l4port
console(config)#dos-control icmp
console(config)#exit
```

Example #2: Viewing the DoS Configuration Information

```
console#show dos-control


SIPDIP Mode..................................... Enable
First Fragment Mode............................. Enable
Min TCP Hdr Size................................ 20
TCP Fragment Mode............................... Enable
TCP Flag Mode................................... Disable
L4 Port Mode.................................... Enable
ICMP Mode....................................... Enable
Max ICMP Pkt Size............................... 512
```

DHCP Snooping

Dynamic Host Configuration Protocol (DHCP) Snooping is a security feature that monitors DHCP messages between a DHCP client and DHCP server to:

- Filter harmful DHCP messages
- Build a bindings database of (MAC address, IP address, VLAN ID, port) authorized tuples.

DHCP snooping is disabled globally and on all VLANs by default. Ports are untrusted by default.

Network administrators can enable DHCP snooping globally and on specific VLANs. They can also configure ports within the VLAN to be trusted or untrusted. DHCP servers must be reached through trusted ports.

DHCP snooping enforces the following security rules:

- DHCP packets from a DHCP server (DHCPOFFER, DHCPACK, DHCPNAK, DHCPRELEASEQUERY) are dropped if received on an untrusted port.
- DHCPRELEASE and DHCPDECLINE messages are dropped if for a MAC addresses in the snooping database, but the binding's interface is other than the interface where the message was received.
- On untrusted interfaces, the switch drops DHCP packets with a source MAC address that does not match the client hardware address. This is a configurable option.

Dynamic ARP Inspection uses the DHCP snooping bindings database to validate ARP packets.

To prevent DHCP packets being used as a DoS attack when DHCP snooping is enabled, the snooping application enforces a rate limit for DHCP packets received on interfaces. DHCP snooping monitors the receive rate on each interface separately. If the receive rate exceeds a configurable limit, DHCP snooping brings down the interface. The user must do “no shutdown” on this interface to further work with that port. The user can configure both the rate and the burst interval.

The hardware rate limits DHCP packets sent to the CPU from interfaces to 64 Kbps.

The DHCP snooping application processes incoming DHCP messages. For DHCPRELEASE and DHCPDECLINE messages, the application compares the receive interface and VLAN with the client interface and VLAN in the bindings database. If the interfaces do not match, the application logs the event and drops the message. For valid client messages, DHCP snooping compares the source MAC address to the DHCP client hardware address. When there is a mismatch, DHCP snooping logs and drops the packet. The network administrator can disable this feature using the **no ip dhcp snooping verify mac-address** command. DHCP snooping forwards valid client messages on trusted members within the VLAN. If DHCP relay co-exists with DHCP snooping, DHCP client messages are sent to DHCP relay for further processing.

The DHCP snooping application uses DHCP messages to build and maintain the binding's database. The binding's database only includes data for clients on untrusted ports. DHCP snooping creates a tentative binding from DHCP DISCOVER and REQUEST messages. Tentative bindings tie a client to a port (the port where the DHCP client message was received). Tentative bindings are completed when DHCP snooping learns the client's IP address from a DHCP ACK message on a trusted port. DHCP snooping removes bindings in response to DECLINE, RELEASE, and NACK messages. DHCP Snooping application ignores the ACK messages as reply to the DHCP Inform messages received on trusted ports. The administrator can also enter static bindings into the binding database.

The DHCP binding database resides on a configured external server or locally in flash depending upon the user configuration. When a switch learns of new bindings or when it loses bindings, the switch immediately updates the entries in the database. The switch also updates the entries in the bindings file. The frequency at which the file is updated is based on a configurable delay, and the updates are batched.

If the absolute lease time of the snooping database entry expires, the entry is removed. If the system time is not consistent across reboots, snooping entries will not expire properly. If a host sends a DHCP release while the switch is rebooting, when the switch receives the DHCP discovery or request, the client's binding will go to the tentative binding.

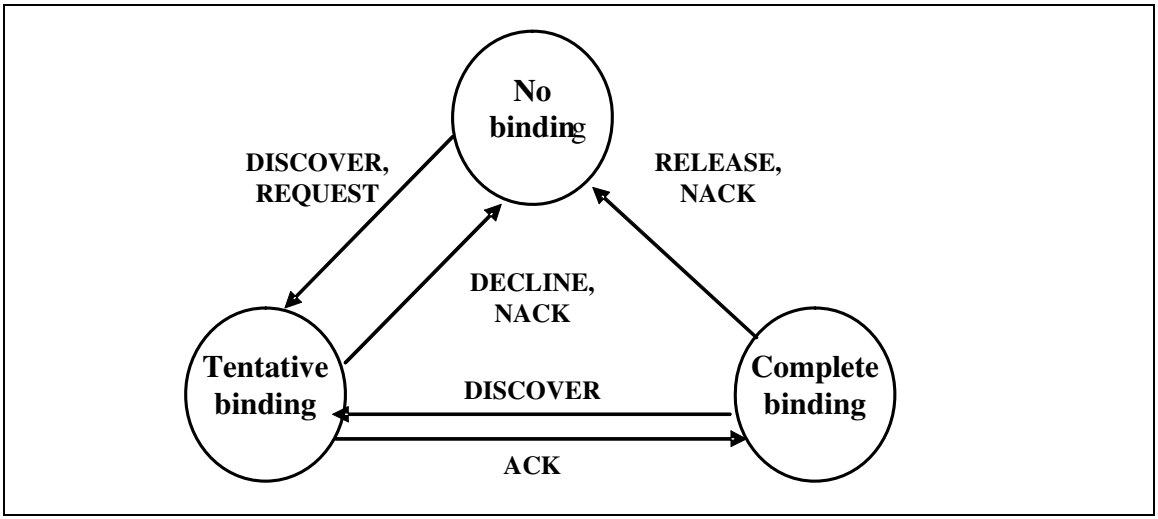


Figure 3-4. DHCP Binding

The DHCP snooping component does not forward server messages since they are forwarded in hardware.

DHCP snooping forwards valid DHCP client messages received on un-trusted interfaces to all trusted interfaces within the VLAN.

The binding's database includes the following information for each entry:

- Client MAC address
- Client IP address
- Time when client lease expires
- Client VLAN ID
- Client port

DHCP snooping can be configured on switching VLANs and routing VLANs. When a DHCP packet is received on a routing VLAN, the DHCP snooping application applies its filtering rules and updates the bindings database. If a client message passes filtering rules, the message is placed into the software forwarding path where it may be processed by the DHCP relay agent or forwarded as an IP packet.

CLI Examples

The commands below show examples of configuring DHCP Snooping for the switch and for individual interfaces.

Example #1 Enable DHCP snooping for the switch

```
console(config)#ip dhcp snooping
console(config)#exit
console#
```

Example #2 Enable DHCP snooping on a VLAN

```
console(config)#ip dhcp snooping vlan 1
console(config)#exit
console#
```

Example #3 Enable DHCP snooping's Source MAC verification

```
console(config)#ip dhcp snooping verify mac-address
console(config)#exit
```

Example #4 Configure DHCP snooping database remote storage parameters

```
console(config)#ip dhcp snooping database tftp://10.131.11.1/dsDb.txt
console(config)#
console(config)#exit
```

Example #5 Configure DHCP snooping database Local storage parameters

```
console(config)#ip dhcp snooping database local
```

```
console(config)#
console(config)#exit
```

Example #6 Configure DHCP snooping database Persistency interval

```
console(config)#ip dhcp snooping database write-delay 500
console(config)#
console(config)#exit
```

Example #7 Configure an interface as DHCP snooping trusted

```
console(config-if-1/g1)#ip dhcp snooping trust
console(config-if-1/g1)#exit
```

Example #8 Configure rate limiting on an interface

```
console(config-if-1/g1)#ip dhcp snooping limit rate 50 burst interval 1
console(config-if-1/g1)#exit
```

Example #9 Configure a DHCP snooping static binding entry

```
console(config)#ip dhcp snooping binding 00:01:02:03:04:05 vlan 1
10.131.11.1 interface 1/g2
console(config)#exit
```

Example #10 Show DHCP Snooping configuration on VLANs and Ports

```
show ip dhcp snooping binding
DHCP snooping is Enabled
DHCP snooping source MAC verification is enabled
DHCP snooping is enabled on the following VLANs:
1

 Interface    Trusted     Log Invalid Pkts
-----------  ----------   ----------------
1/g1         Yes          Yes
1/g2         No           No
1/g3         No           No
1/g4         No           No
1/g5         No           No
1/g6         No           No
1/g7         No           No
1/g8         No           No
1/g9         No           No
1/g10        No           No
1/g11        No           No
1/g12        No           No
1/g13        No           No
1/g14        No           No


--More-- or (q)uit

 Interface    Trusted     Log Invalid Pkts
```

```
-----------  ----------   ----------------
1/g15        No           No
1/g16        No           No
1/g17        No           No
1/g18        No           No
1/g19        No           No
1/g20        No           No
1/g21        No           No
1/g22        No           No
1/g23        No           No
1/g24        No           No
1/xg3        No           No
1/xg4        No           No
ch1          No           No
ch2          No           No
ch3          No           No
ch4          No           No
ch5          No           No
ch6          No           No

--More-- or (q)uit

console#
```

Example #12 Show DHCP Snooping database configurations

```
console#show ip dhcp snooping database
agent url:  local

write-delay:  500


console#
```

Example #13 Show DHCP Snooping binding entries

```
Total number of bindings:  2

   MAC Address       IP Address     VLAN   Interface    Type    Lease (Secs)
-----------------  ---------------  ----  -----------  -------  -----------
00:01:02:03:04:05      10.131.11.1     1         1/g2   STATIC
00:02:B3:06:60:80      10.131.11.3     1         1/g2  DYNAMIC  86400
```

Example #14 Show DHCP Snooping Per Port rate limiting configurations

```
show ip dhcp snooping interfaces

Interface     Trust State     Rate Limit      Burst Interval
                                 (pps)           (seconds)
----------   -------------   -------------   ---------------

1/g1         Yes             50              1
1/g2         No              15              1
```

```
1/g3          No              15              1
1/g4          No              15              1
1/g5          No              15              1
1/g6          No              15              1
1/g7          No              15              1
1/g8          No              15              1
1/g9          No              15              1
1/g10         No              15              1
1/g11         No              15              1
1/g12         No              15              1
1/g13         No              15              1
1/g14         No              15              1
1/g15         No              15              1
1/g16         No              15              1
1/g17         No              15              1
1/g18         No              15              1

--More-- or (q)uit

1/g19         No              15              1
1/g20         No              15              1
1/g21         No              15              1
1/g22         No              15              1
1/g23         No              15              1
1/g24         No              15              1
1/xg3         No              15              1
1/xg4         No              15              1
ch1           No              15              1
ch2           No              15              1
```

```
ch3          No              15              1
ch4          No              15              1
ch5          No              15              1
ch6          No              15              1
ch7          No              15              1
ch8          No              15              1
ch9          No              15              1
ch10         No              15              1

--More-- or (q)uit

console#
```

Example #15 Show DHCP Snooping Per Port Statistics

```
console#show ip dhcp snooping statistics

 Interface    MAC Verify   Client Ifc   DHCP Server
               Failures     Mismatch    Msgs Rec'd
-----------   ----------   ----------   -----------
1/g2                   0            0             0
1/g3                   0            0             0
1/g4                   0            0             0
1/g5                   0            0             0
1/g6                   0            0             0
1/g7                   0            0             0
1/g8                   0            0             0
1/g9                   0            0             0
1/g10                  0            0             0
```

```
1/g11                0              0              0
1/g12                0              0              0
1/g13                0              0              0
1/g14                0              0              0
1/g15                0              0              0
1/g16                0              0              0
1/g17                0              0              0
1/g18                0              0              0
1/g19                0              0              0
1/g20                0              0              0
--More-- or (q)uit
1/g21                0              0              0
1/g22                0              0              0
1/g23                0              0              0
1/g24                0              0              0
1/xg3                0              0              0
1/xg4                0              0              0
ch1                  0              0              0
ch2                  0              0              0
ch3                  0              0              0
ch4                  0              0              0
ch5                  0              0              0
ch6                  0              0              0
ch7                  0              0              0
ch8                  0              0              0
ch9                  0              0              0
ch10                 0              0              0
ch11                 0              0              0
ch12                 0              0              0
```

```
ch13                    0               0               0
ch14                    0               0               0
ch15                    0               0               0
ch16                    0               0               0
ch17                    0               0               0
--More-- or (q)uit
```

sFlow

This section describes the sFlow feature. sFlow is the industry standard for monitoring high-speed switched and routed networks. sFlow technology is built into network equipment and gives complete visibility into network activity, enabling effective management and control of network resources.

Overview

As illustrated in Figure 3-5, the sFlow monitoring system consists of sFlow Agents (embedded in a switch, router, or standalone probe) and a central sFlow Collector. sFlow Agents use sampling technology to capture traffic statistics from monitored devices. sFlow datagrams forward sampled traffic statistics to the sFlow Collector for analysis.

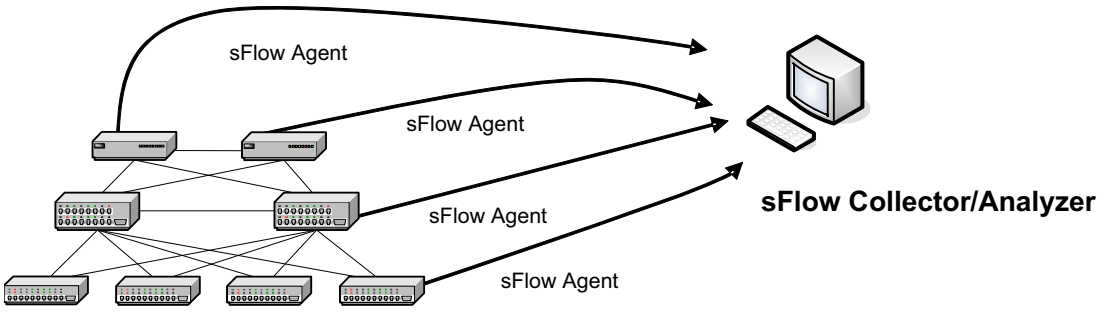


Figure 3-5. sFlow Architecture

The advantages of using sFlow are:

- It is possible to monitor all ports of the switch continuously, with no impact on the distributed switching performance.
- Minimal memory/CPU is required. Samples are not aggregated into a flow-table on the switch; they are forwarded immediately over the network to the sFlow collector.
- System is tolerant to packet loss in the network (statistical model means loss is equivalent to slight change in sampling rate).
- sFlow collector can receive data from multiple switches, providing a real-time synchronized view of the whole network.
- The Collector can analyze traffic patterns based on protocols found in the headers (e.g., TCP/IP, IPX, Ethernet, AppleTalk…). This alleviates the need for a layer 2 switch to decode and understand all protocols.

sFlow Agents

sFlow Agents use two forms of sampling:

- Statistical packet-based sampling of switched or routed Packet Flows
- Time-based sampling of counters

Packet Flow Sampling and Counter Sampling are performed by sFlow Instances associated with individual Data Sources within an sFlow Agent. Both types of samples are combined in sFlow datagrams. Packet Flow Sampling creates a steady, but random, stream of sFlow datagrams that are sent to the sFlow Collector. Counter samples may be taken opportunistically to fill these datagrams.

To perform Packet Flow Sampling, an sFlow Sampler Instance is configured with a Sampling Rate. Packet Flow sampling results in the generation of Packet Flow Records. To perform Counter Sampling, an sFlow Poller Instance is configured with a Polling Interval. Counter Sampling results in the generation of Counter Records. sFlow Agents collect Counter Records and Packet Flow Records and send them as sFlow datagrams to sFlow Collectors.

Packet Flow Sampling

Packet Flow Sampling, carried out by each sFlow instance, ensures that any packet observed at a Data Source has an equal chance of being sampled, irrespective of the Packet Flow(s) to which it belongs.

Packet Flow Sampling is accomplished as follows:

1. A packet arrives on an interface.
2. The Network Device makes a filtering decision to determine whether the packet should be dropped.
3. If the packet is not filtered (dropped) a destination interface is assigned by the switching/routing function.
4. A decision is made on whether or not to sample the packet.

The mechanism involves a counter that is decremented with each packet. When the counter reaches zero a sample is taken.

5. When a sample is taken, the counter indicating how many packets to skip before taking the next sample is reset. The value of the counter is set to a random integer where the sequence of random integers used over time is the Sampling Rate.

Counter Sampling

The primary objective of Counter Sampling is to efficiently, periodically export counters associated with Data Sources. A maximum Sampling Interval is assigned to each sFlow instance associated with a Data Source.

Counter Sampling is accomplished as follows:

- sFlow Agents keep a list of counter sources being sampled.
- When a Packet Flow Sample is generated the sFlow Agent examines the list and adds counters to the sample datagram, least recently sampled first. Counters are only added to the datagram if the sources are within a short period, 5 seconds say, of failing to meet the required Sampling Interval.
- Periodically, say every second, the sFlow Agent examines the list of counter sources and sends any counters that must be sent to meet the sampling interval requirement.

The set of counters is a fixed set.

CLI Examples

The following are examples of the commands used for sFlow.

Example #1: Configure destination IP address and maxdatagram size for an sFlow receiver index

```
console(config)#sflow 1 destination 30.30.30.1 560
```

Example #2: Configure sFlow on an Ethernet interface range with a polling interval of 200 seconds

```
console(config)#sflow 1 polling ethernet 1/g1-1/g10 200
```

Example #3: Configure sFlow on an Ethernet interface with a polling interval of 400 seconds

```
console(config-if-1/g15)#sflow 1 polling 400
```

Example #4: Show the sFlow configuration for receiver index 1

```
console#show sflow 1 destination

Receiver Index.............................. 1
Owner String................................ site77
Time out.................................... 1529
IP Address:................................. 30.30.30.1
Address Type................................ 1
Port........................................ 560
Datagram Version............................ 5
Maximum Datagram Size....................... 500

Example #5: Show sFlow sampling for receiver index 1
console#show sflow 1 sampling

 Sampler      Receiver  Packet          Max Header
 Data Source  Index     Sampling Rate   Size
 -----------  -------   -------------   ----------
 1/g1         1         1500            50
 1/g2         1         1500            50
 1/g3         1         1500            50
 1/g4         1         1500            50
 1/g5         1         1500            50
 1/g6         1         1500            50
 1/g7         1         1500            50
 1/g8         1         1500            50
 1/g9         1         1500            50
 1/g10        1         1500            50
 1/g15        1         1500            50
```

Example #6: Show sFlow polling for receiver index 1

```
console#show sflow 1 polling

 Poller         Receiver  Poller
 Data Source    Index     Interval
 -----------    -------   -------
 1/g1           1         200
 1/g2           1         200
 1/g3           1         200
 1/g4           1         200
 1/g5           1         200
 1/g6           1         200
 1/g7           1         200
 1/g8           1         200
 1/g9           1         200
 1/g10          1         200
 1/g15          1         400
```

4

Routing Configuration

This section describes configuration scenarios and instructions for the following routing features:

- "VLAN Routing" on page 74
- "Virtual Router Redundancy Protocol" on page 77
- "Proxy Address Resolution Protocol (ARP)" on page 80
- "OSPF" on page 81
- "Routing Information Protocol" on page 92
- "Route Preferences" on page 95
- "Loopback Interfaces" on page 99
- "IP Helper" on page 100

VLAN Routing

This section provides an example of how to configure PowerConnect 6200 Series software to support VLAN routing.

NOTE: The management VLAN cannot be configured as a routing interface. The switch may also be managed via VLAN routing interfaces.

CLI Examples

The diagram in this section shows a Layer 3 switch configured for VLAN routing. It connects two VLANs, with two ports participating in one VLAN, and one port in the other. The script shows the commands you would use to configure PowerConnect 6200 Series software to provide the VLAN routing support shown in the diagram.

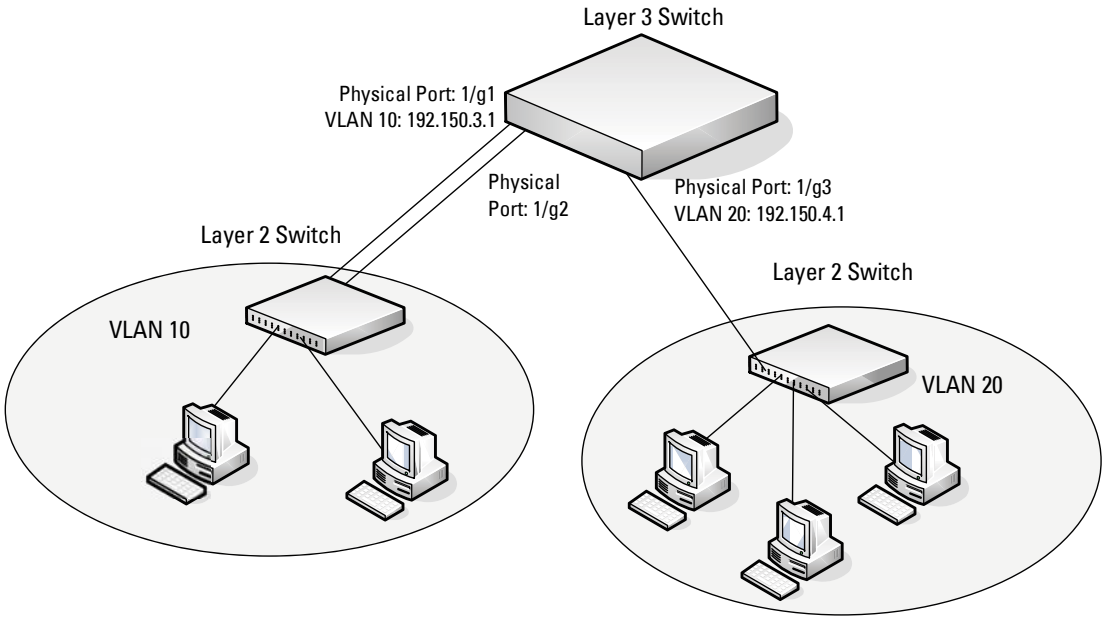


Figure 4-1. VLAN Routing Example Network Diagram

Example 1: Create Two VLANs

The following code sequence shows an example of creating two VLANs with egress frame tagging enabled.

```
console#configure

console(config)#vlan database
```

```
console(config-vlan)#vlan 10
console(config-vlan)#vlan 20
console(config-vlan)#exit
```

Example 2: Configure the VLAN Members

The following code sequence shows an example of adding ports to the VLANs and assigning the PVID for each port. The PVID determines the VLAN ID assigned to untagged frames received on the ports.

```
console#configure
console(config)#interface ethernet 1/g1
console(config-if-1/g1)#switchport mode general
console(config-if-1/g1)#switchport general allowed vlan add 10
console(config-if-1/g1)#switchport general pvid 10
console(config-if-1/g1)#exit

console#configure
console(config)#interface ethernet 1/g2
console(config-if-1/g2)#switchport mode general
console(config-if-1/g2)#switchport general allowed vlan add 10
console(config-if-1/g2)#switchport general pvid 10
console(config-if-1/g2)#exit

console#configure
console(config)#interface ethernet 1/g3
console(config-if-1/g3)#switchport mode general
console(config-if-1/g3)#switchport general allowed vlan add 20
console(config-if-1/g3)#switchport general pvid 20
console(config-if-1/g3)#exit
```

Example 3: Set Up VLAN Routing for the VLANs and Assign an IP Address

The following code sequence shows how to enable routing for the VLANs and how to configure the IP addresses and subnet masks for the virtual router ports.:

```
console#configure
console(config)#interface vlan 10
console(config-if-vlan10)#routing
console(config-if-vlan10)#ip address 192.150.3.1 255.255.255.0
console(config-if-vlan10)#exit
console#configure
console(config)#interface vlan 20
console(config-if-vlan20)#routing
console(config-if-vlan20)#ip address 192.150.4.1 255.255.255.0
console(config-if-vlan20)#exit
```

Example 4: Enable Routing for the Switch:

In order for the VLAN to function as a routing interface, you must enable routing on the VLAN and on the switch.

```
console(config)#ip routing
```

Using the Web Interface to Configure VLAN Routing

Use the following screens to perform the same configuration using the Web Interface:

- **Switching > VLAN > VLAN Membership**. To create the VLANs and specify port participation.
- **Switching > VLAN > Port Settings**. To set the PVID and VLAN type.
- **Routing > VLAN Routing > Configuration.** To enable routing on Vlans.
- **Routing > IP > Configuration**. To enable routing for the switch.
- **Routing > IP > Interface Configuration**. To configure VLAN IP addresses and subnet masks.

Virtual Router Redundancy Protocol

When an end station is statically configured with the address of the router that will handle its routed traffic, a single point of failure is introduced into the network. If the router goes down, the end station is unable to communicate. Since static configuration is a convenient way to assign router addresses, Virtual Router Redundancy Protocol (VRRP) was developed to provide a backup mechanism.

VRRP eliminates the single point of failure associated with static default routes by enabling a *backup* router to take over from a *master* router without affecting the end stations using the route. The end stations will use a *virtual* IP address that is recognized by the backup router if the master router fails. Participating routers use an election protocol to determine which router is the master router at any given time. A given VLAN routing interface may appear as more than one virtual router to the network. Also, more than one VLAN routing interface on a switch may participate in a virtual router.

CLI Examples

This example shows how to configure the switch to support VRRP. Router 1 is the default master router for the virtual route, and Router 2 is the backup router.

NOTE: The VRRP IP address on a routing interface must belong to the same subnet (primary or secondary) as the IP address (primary or secondary) configured on that routing interface; otherwise, an error message displays and the VRRP IP configuration fails. The master and backup VLAN routing interfaces must be in the same subnet and be members of the same VLAN.

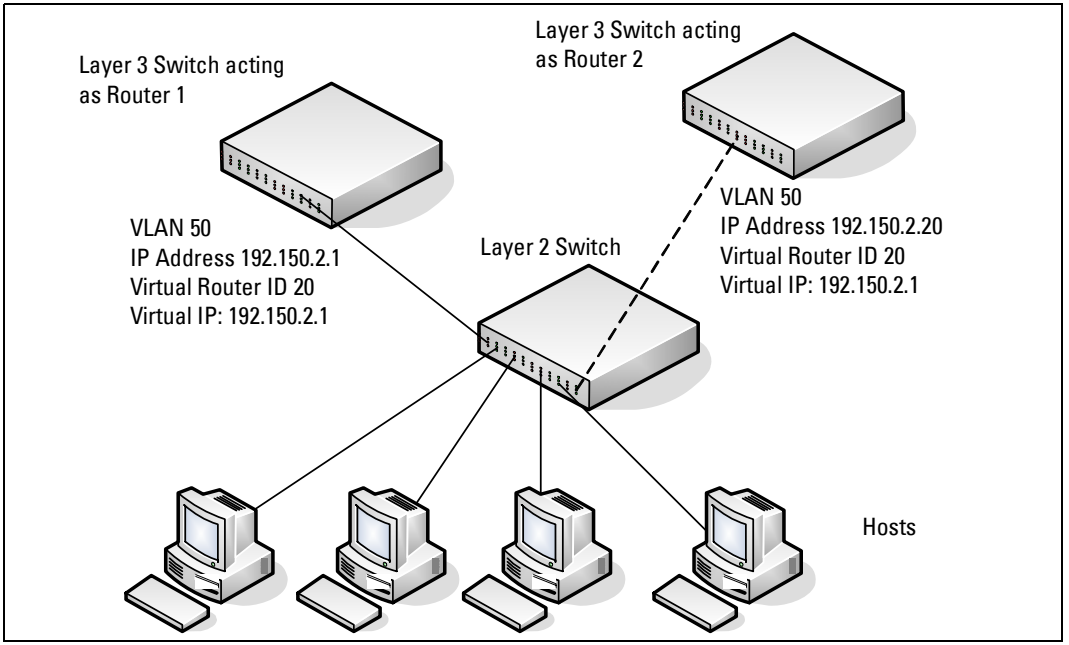


Figure 4-2. VRRP Example Network Configuration

Configuring VRRP on the Switch as a Master Router

1. Enable routing for the switch. IP forwarding is then enabled by default.

```
console#config
console(config)#ip routing
```

2. Configure the IP addresses and subnet masks for the VLAN routing interface that will participate in the protocol:

```
console(config)#interface vlan 50
console(config-if-vlan50)#ip address 192.150.2.1 255.255.255.0
console(config-if-vlan50)#exit
```

3. Enable VRRP for the switch:

```
console(config)#ip vrrp
```

4. Assign the virtual router ID to the interface that will participate in the protocol:

```
console(config)#interface vlan 50
console(config-if-vlan50)#ip vrrp 20
```

5. Specify the IP address that the virtual router function will recognize. The interface IP address is the same as the virtual IP address. This means the router is the *interface owner* and therefore has a priority of 255, which guarantees that it is the master.

```
console(config-if-vlan50)#ip vrrp 20 ip 192.150.2.1
```

6. Start the virtual router on the interface:

```
console(config-if-vlan50)#ip vrrp 20 mode
console(config-if-vlan50)#exit
```

Configuring VRRP on the Switch as a Backup Router

1. Enable routing for the switch. IP forwarding is then enabled by default.

```
console#config
console(config)#ip routing
```

2. Configure the IP addresses and subnet masks for the port that will participate in the protocol:

```
console(config)#interface vlan 50
console(config-if-vlan50)#ip address 192.150.2.20 255.255.255.0
console(config-if-vlan50)#exit
```

3. Enable VRRP for the switch.

```
console(config)#ip vrrp
```

4 Assign virtual router ID to the interface that will participate in the protocol:

```
console(config)#interface vlan 50

console(config-if-vlan50)#ip vrrp 20
```

5 Specify the IP address that the virtual router function will recognize.

```
console(config-if-vlan50)#ip vrrp 20 ip 192.150.2.1
```

6 Set the priority for the interface. Assigning a lower priority value than the interface on the other router ensures that this interface the backup.

```
console(config-if-vlan50)#ip vrrp 20 priority 250
```

7 Start the virtual router on the interface.

```
console(config-if-vlan50)#ip vrrp 20 mode

console(config-if-vlan50)#exit
```

Using the Web Interface to Configure VRRP

Use the following screens to perform the same configuration using the Graphical User Interface:

- **Routing > IP > Configuration.** To enable routing for the switch.
- **Routing > IP > Interface Configuration.** To enable routing for the VLAN interfaces and configure their IP addresses and subnet masks.
- **Routing > VRRP > VRRP Configuration.** To enable VRRP for the switch
- **Routing > VRRP > Virtual Router Configuration.** To configure the interface for VRRP.

Proxy Address Resolution Protocol (ARP)

This section describes the Proxy Address Resolution Protocol (ARP) feature.

Overview

- Proxy ARP allows a router to answer ARP requests where the target IP address is not the router itself but a destination that the router can reach.
- If a host does not know the default gateway, proxy ARP can learn the first hop.
- Machines in one physical network appear to be part of another logical network.
- Without proxy ARP, a router responds to an ARP request only if the target IP address is an address configured on the interface where the ARP request arrived.

CLI Examples

The following are examples of the commands used in the proxy ARP feature.

Example #1: Enabling Proxy ARP

To enable IP Proxy ARP:

```
console#config

console(config)#interface vlan 10

console(config-if-vlan10)#routing

console(config-if-vlan10)#ip proxy-arp

console(config-if-vlan10)#exit
```

Example #2 Viewing the Interface Information

```
console#show ip interface vlan 50


Primary IP Address.............................
192.150.2.1/255.255.255.0

Routing Mode..................................... Enable

Administrative Mode.............................. Enable

Forward Net Directed Broadcasts.................. Disable

Proxy ARP........................................ Enable

Local Proxy ARP.................................. Disable
```

```
Active State..................................... Inactive
Link Speed Data Rate............................. 10 Half
MAC Address...................................... 00FF.F2A3.888A
Encapsulation Type............................... Ethernet
IP MTU........................................... 1500
```

OSPF

Larger networks typically use the Open Shortest Path First (OSPF) protocol instead of RIP. To the administrator of a large and/or complex network, OSPF offers several benefits:

- Less network traffic:
  - Routing table updates are sent only when a change has occurred.
  - Only the part of the table that has changed is sent.
  - Updates are sent to a multicast, not a broadcast address.
- Hierarchical management: allows the network to be subdivided.

The switch supports OSPFv2, which is used on IPv4 networks and OSPFv3, which has enhancements for handling 128-bit IPv6 addresses. The protocols are configured separately within the software, but their functionality is largely similar for IPv4 and IPv6 networks. The following description applies to both protocols, except where noted.

OSPF Concepts and Terms

Figure 4-3, Figure 4-4, and Figure 4-5 show example OSPF topologies that illustrate the concepts described in this section.

Areas and Topology

The top level of the hierarchy of an OSPF network is known as an autonomous system (AS) or routing domain, and is a collection of networks with a common administration and routing strategy. The AS is divided into *areas*. Routers within an area must share detailed information on the topology of their area, but require less detailed information about the topology of other areas. Segregating a network into areas enables limiting the amount of route information communicated throughout the network.

Areas are identified by a numeric ID in IP address format *n.n.n.n* (note, however, that these are not used as actual IP addresses). For simplicity, the area can be configured and referred to in normal integer notation; however, the software converts these to dot notation by using the right-most octet up to 255 and proceeding to the next left octet for higher values (i.e., Area 20 is identified as 0.0.0.20 and Area 256 as 0.0.1.0). The area identified as 0.0.0.0 is referred to as *Area 0* and is considered the *OSPF backbone*. All other OSPF areas in the network must connect to Area 0 directly or through a virtual link. The backbone area is responsible for distributing routing information between non-backbone areas.

A *virtual link* can be used to connect an area to Area 0 when a direct link is not possible. A virtual link traverses an area between the remote area and Area 0 (see Figure 4-5).

A *stub area* is an area that does not receive routes that were learned from a protocol other than OSPF or were statically configured. These routes typically send traffic outside the AS. Therefore, routes from a stub area to locations outside the AS use the default gateway. A virtual link cannot be configured across a stub area. A *Not So Stubby Area* can import limited external routes only from a connected ASBR.

OSPF Routers and LSAs

OSPF routers keep track of the state of the various links they send data to. Routers share OSPF *link state advertisements* (LSAs) with other routers. Various LSA types provide detailed information on a link for sharing within an area or summary information for sharing outside an area. External LSAs provide information on static routes or routes learned from other routing protocols.

OSPF defines various router types:

- Backbone routers have an interface in Area 0. They condense and summarize information about all the areas in the AS and advertise this information on the backbone.
- Area border routers (ABRs) connect areas to the OSPF backbone (in the case of virtual links, the an ABR may connect to another ABR that provides a direct connection to Area 0). An ABR is a member of each area it connects to.
- Internal routers (IRs) route traffic within an area. When two routers in an area discover each other through OSPF Hello messages, they are called OSPF neighbors. Neighbors share detailed information on the topology of the area using local LSAs.
- Autonomous system boundary routers (ASBRs) connect to other ASes. ASBRs use other protocols such as BGP or RIP to communicate outside the AS. The ASBR performs route redistribution; i.e., when it learns routes from other protocols, it originates external LSAs that advertise those prefixes within the AS.

Metrics and Route Selection

You can configure the metric type of external routes originated through route redistribution. The metric type influences the routes computed by other OSPF routers in the domain.

OSPF determines the best route using the assigned cost and the type of the OSPF route. The following order is used for choosing a route if more than one type of route exists:

1. Intra-area (the source and destination address are in the same area)
2. Inter-area (the source and destination are not in the same area, i.e., the route crosses the OSPF backbone)
3. External Type 1
4. External Type 2

External routes are those imported into OSPF from other routing protocol or processes. OSPF computes the path cost differently for external type 1 and external type 2 routes. The cost of an external type 1 route is the cost advertised in the external LSA plus the path cost from the calculating router to the ASBR. The cost of an external type 2 route is the cost advertised by the ASBR in its external LSA.

NOTE: The following example uses the CLI to configure OSPF. You can also use the Web interface. Click **Routing > OSPF** or **IPv6 > OSPFv3** in the navigation tree.

CLI Examples

Example 1: Configuring an OSPF Border Router and Setting Interface Costs

The following example shows you how to configure an OSPF border router areas and interfaces in the switch.

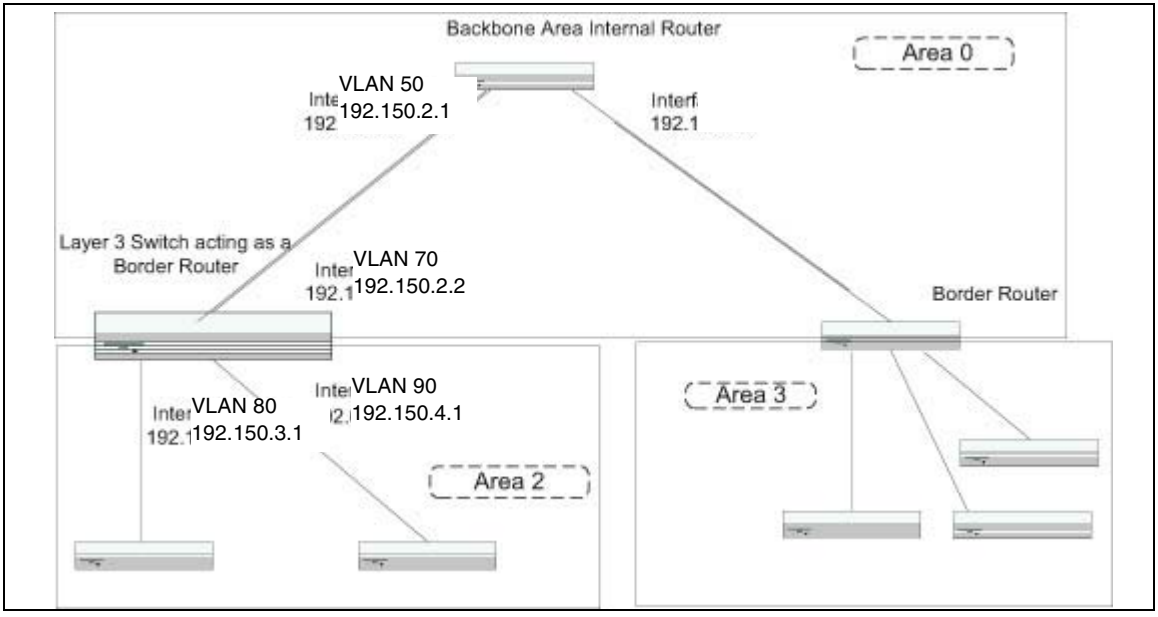


Figure 4-3. OSPF Example Network Diagram: Border Router

IPv4 (OSPFv2)	IPv6 (OSPFv3)
• Enable routing for the switch:	

`console#config`
`ip routing`
`exit`
	`console#config`
	`ipv6 unicast-routing`
	`exit`

Enable routing and assign IP for VLANs 70, 80 and 90.	

`config`
`interface vlan 70`
`routing`
`ip address 192.150.2.2 255.255.255.0`
`exit`
`interface vlan 80`
`routing`
`ip address 192.130.3.1 255.255.255.0`
`exit`
`interface vlan 90`
`routing`
`ip address 192.64.4.1 255.255.255.0`
`exit`
`exit`
	`config`
	`interface vlan 70`
	`routing`
	`ipv6 enable`
	`exit`
	`interface vlan 80`
	`routing`
	`ipv6 address 2002::1/64`
	`exit`
	`interface vlan 90`
	`routing`
	`ipv6 address 2003::1/64`
	`exit`
	`exit`

Specify a router ID. Disable 1583 compatibility to prevent a routing loop (IPv4-only).	

`config`
`router ospf`
`router-id 192.150.9.9`
`no 1583compatibility`
`exit`
`exit`
	`config`
	`ipv6 router ospf`
	`router-id 1.1.1.1`
	`exit`
	`exit`

OSPF is globally enabled by default. To make it operational on the router, you configure OSPF for particular interfaces and identify which area the interface is associated with. The following commands also sets the priority and cost for the ports:

IPv4 (OSPFv2)	IPv6 (OSPFv3)
config interface vlan 70 ip ospf area 0.0.0.0 ip ospf priority 128 ip ospf cost 32 exit interface vlan 80 ip ospf area 0.0.0.2 ip ospf priority 255 ip ospf cost 64 exit interface vlan 90 ip ospf area 0.0.0.2 ip ospf priority 255 ip ospf cost 64 exit exit	config interface vlan 70 ipv6 ospf ipv6 ospf areaid 0.0.0.0 ipv6 ospf priority 128 ipv6 ospf cost 32 exit interface vlan 80 ipv6 ospf ipv6 ospf areaid 0.0.0.2 ipv6 ospf priority 255 ipv6 ospf cost 64 exit interface vlan 90 ipv6 ospf ipv6 ospf areaid 0.0.0.2 ipv6 ospf priority 255 ipv6 ospf cost 64 exit exit

Example 2: Configuring Stub and NSSA Areas

In this example, Area 0 connects directly to two other areas: Area 1 is defined as a stub area and Area 2 is defined as an NSSA area.

NOTE: OSPFv2 and OSPFv3 can operate concurrently on a network and on the same interfaces (although they do not interact). This example configures both protocols simultaneously.

Figure 4-4 illustrates this example OSPF configuration.

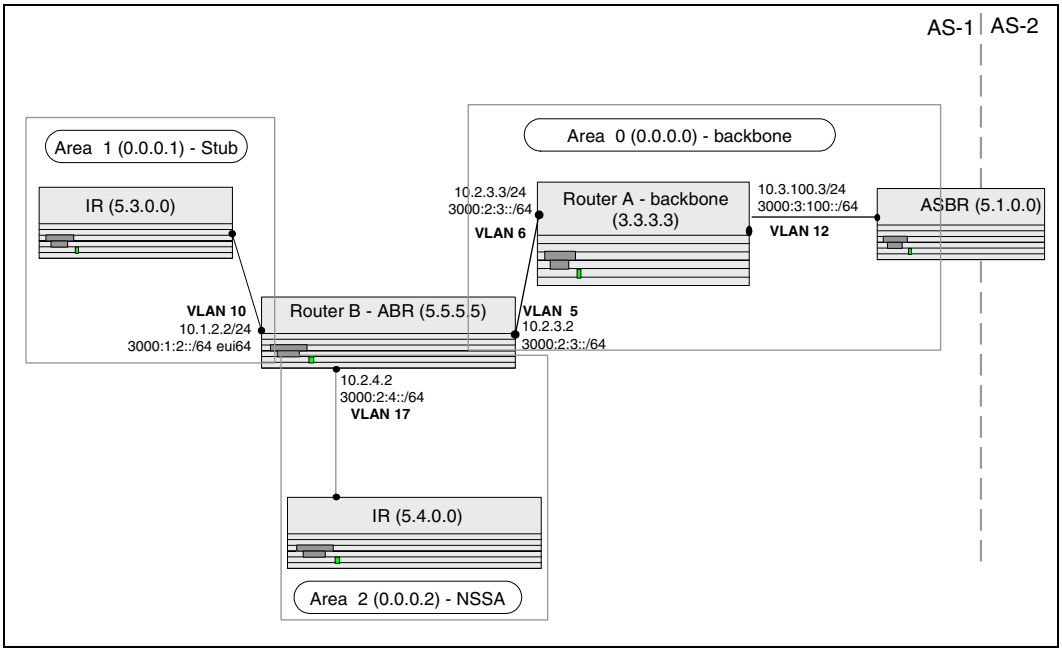


Figure 4-4. OSPF Configuration—Stub Area and NSSA Area

Configure Router A: Router A is a backbone router. It links to an ASBR (not defined here) that routes traffic outside the AS.

- Globally enable IPv6 and IPv4 routing:

```
(console) #configure
   ipv6 unicast-routing
   ip routing
```

- Configure IP address and enable OSPF on VLAN routing interfaces 6 and 12 and enable IPv6 OSPF on the interfaces. (OSPF is enabled on the IPv4 interface in the next code group.)

```
interface vlan 6
   routing
   ip address 10.2.3.3  255.255.255.0
   ipv6 address 3000:2:3::/64 eui64
   ip ospf area 0.0.0.0
   ipv6 ospf
   exit


interface vlan 12
   routing
   ip address  10.3.100.3  255.255.255.0
```

```
   ipv6 address 3000:3:100::/64 eui64
   ip ospf area 0.0.0.0
   ipv6 ospf
   exit
```

- Define an OSPF router:

```
ipv6 router ospf
   router-id 3.3.3.3
   exit
router ospf
   router-id 3.3.3.3
   exit
exit
```

Configure Router B: Router B is a ABR that connects Area 0 to Areas 1 and 2.

- Configure IPv6 and IPv4 routing. The static routes are included for illustration only: Redistributed static routes, like routes distributed from other protocols, are not injected into stub areas such as Area 1:

```
(console)#configure
   ipv6 unicast-routing
   ipv6 route 3000:44:44::/64 3000:2:3::210:18ff:fe82:c14
   ip route 10.23.67.0 255.255.255.0 10.2.3.3
```

- On VLANs 10, 5, and 17, configure IPv4 and IPv6 addresses and enable OSPF. For IPv6, associate VLAN 10 with Area 1 and VLAN 17 with Area 2. (OSPF is enabled on the IPv4 VLAN routing interface in the next code group.)

```
interface vlan 10
   routing
   ip address  10.1.2.2  255.255.255.0
   ipv6 address 3000:1:2::/64 eui64
   ipv6 ospf
   ipv6 ospf areaid 1
   exit
interface vlan 5
   routing
   ip address  10.2.3.2  255.255.255.0
   ipv6 address 3000:2:3::/64 eui64
   ipv6 ospf
   exit
interface vlan 17
   routing
   ip address  10.2.4.2  255.255.255.0
   ipv6 address 3000:2:4::/64 eui64
   ipv6 ospf
   ipv6 ospf areaid 2
   exit
```

- For IPv4: Define an OSPF router. Define Area 1 as a stub. Enable OSPF for IPv4 on VLANs 10, 5, and 17 by globally defining the range of IP addresses associated with each interface, and then associating those ranges with Areas 1, 0, and 17, respectively. Then, configure a metric cost to associate with static routes when they are redistributed via OSPF:

```
router ospf
   router-id 2.2.2.2
   area 0.0.0.1 stub
   area 0.0.0.2 nssa
   network 10.1.2.0 0.0.0.255 area 0.0.0.1
   network 10.2.3.0 0.0.0.255 area 0.0.0.0
   network 10.2.4.0 0.0.0.255 area 0.0.0.2
   redistribute static metric 1 subnets
   exit
```

- For IPv6: Define an OSPF router. Define Area 1 as a stub and area 2 as a Not-So-Stubby-Area (NSSA). Configure a metric cost to associate with static routes when they are redistributed via OSPF:

```
ipv6 router ospf
   router-id 2.2.2.2
   area 0.0.0.1 stub
   area 0.0.0.2 nssa
   redistribute static metric 105 metric-type 1
   exit
exit
```

Example 3: Configuring a Virtual Link

In this example, Area 0 connects directly to Area 1. A virtual link is defined that traverses Area 1 and connects to Area 2.

Figure 4-5 illustrates this example OSPF configuration.

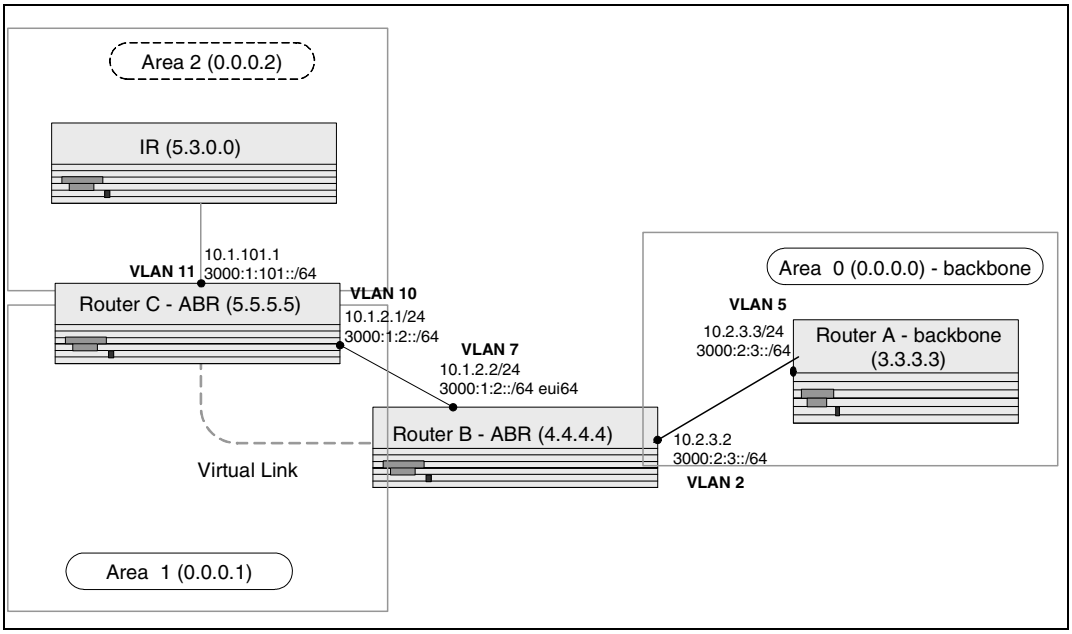


Figure 4-5. OSPF Configuration—Virtual Link

Configure Router A: Router A is a backbone router. Configuration steps are similar to those for Router A in the previous example.

```
(console)#configure
   ipv6 unicast-routing
   ip routing
   exit

   ipv6 router ospf
      router-id 3.3.3.3
      exit

   interface vlan 5
      routing
      ip address  10.2.3.3  255.255.255.0
      ipv6 address 3000:2:3::/64 eui64
      ipv6 ospf
      exit
```

```
router ospf
   router-id 3.3.3.3
   network 10.2.3.0 0.0.0.255 area 0.0.0.0
   exit
exit
```

Configure Router B: Router B is a ABR that directly connects Area 0 to Area 1. In addition to the configuration steps described in the previous example, we define a virtual link that traverses Area 1 to Router C (5.5.5.5).

```
(console)#configure
   ipv6 unicast-routing
   ip routing

   interface vlan 2
      routing
      ip address  10.2.3.2  255.255.255.0
      ipv6 address 3000:2:3::/64 eui64
      ipv6 ospf
      exit

   interface vlan 7
      routing
      ip address  10.1.2.2  255.255.255.0
      ipv6 address 3000:1:2::211:88FF:FE2A:3CB3/64 eui64
      ipv6 ospf
      ipv6 ospf areaid 1
      exit

   router ospf
      router-id 4.4.4.4
      area 0.0.0.1 virtual-link 5.5.5.5
      network 10.2.3.0 0.0.0.255 area 0.0.0.0
      network 10.1.2.0 0.0.0.255 area 0.0.0.1
      exit

   ipv6 router ospf
      router-id 4.4.4.4
      area 0.0.0.1 virtual-link 5.5.5.5
      exit
   exit
```

Configure Router C: Router C is a ABR that enables a virtual link from the remote Area 2 in the AS to Area 0. In addition to the configuration steps described for Router C in the previous example, we define a virtual link that traverses Area 1 to Router B (4.4.4.4).

```
(console)#configure
   ipv6 unicast-routing
   ip routing

   interface vlan 10
```

```
    routing
    ip address  10.1.2.1  255.255.255.0
    ipv6 address 3000:1:2::/64 eui64
    ipv6 ospf
    ipv6 ospf areaid 1
    exit
interface vlan 11
    routing
    ip address 10.1.101.1 255.255.255.0
    ipv6 address 3000:1:101::/64 eui64
    ipv6 ospf
    ipv6 ospf areaid 2
    exit
ipv6 router ospf
    router-id 5.5.5.5
    area 0.0.0.1 virtual-link 4.4.4.4
    exit
router ospf
    router-id 5.5.5.5
    area 0.0.0.1 virtual-link 4.4.4.4
    network 10.1.2.0 0.0.0.255 area 0.0.0.1
    network 10.1.101.0 0.0.0.255 area 0.0.0.2
    exit
exit
```

Routing Information Protocol

Routing Information Protocol (RIP) is one of the protocols which may be used by routers to exchange network topology information. It is characterized as an "interior" gateway protocol, and is typically used in small to medium-sized networks.

RIP Configuration

A router running RIP sends the contents of its routing table to each of its adjacent routers every 30 seconds. When a route is removed from the routing table it is flagged as unusable by the receiving routers after 180 seconds, and removed from their tables after an additional 120 seconds.

There are two versions of RIP:

- RIP-1 defined in RFC 1058
  - Routes are specified by IP destination network and hop count
  - The routing table is broadcast to all stations on the attached network
- RIP-2 defined in RFC 1723
  - Route specification is extended to include subnet mask and gateway
  - The routing table is sent to a multicast address, reducing network traffic
  - An authentication method is used for security

The PowerConnect 6200 Series supports both versions of RIP. You may configure a given port:

- To receive packets in either or both formats
- To transmit packets formatted for RIP-1 or RIP-2 or to send RIP-2 packets to the RIP-1 broadcast address
- To prevent any RIP packets from being received
- To prevent any RIP packets from being transmitted

CLI Examples

The configuration commands used in the following example enable RIP on ports vlan 2 and vlan 3 as shown in the network illustrated in Figure 4-6.

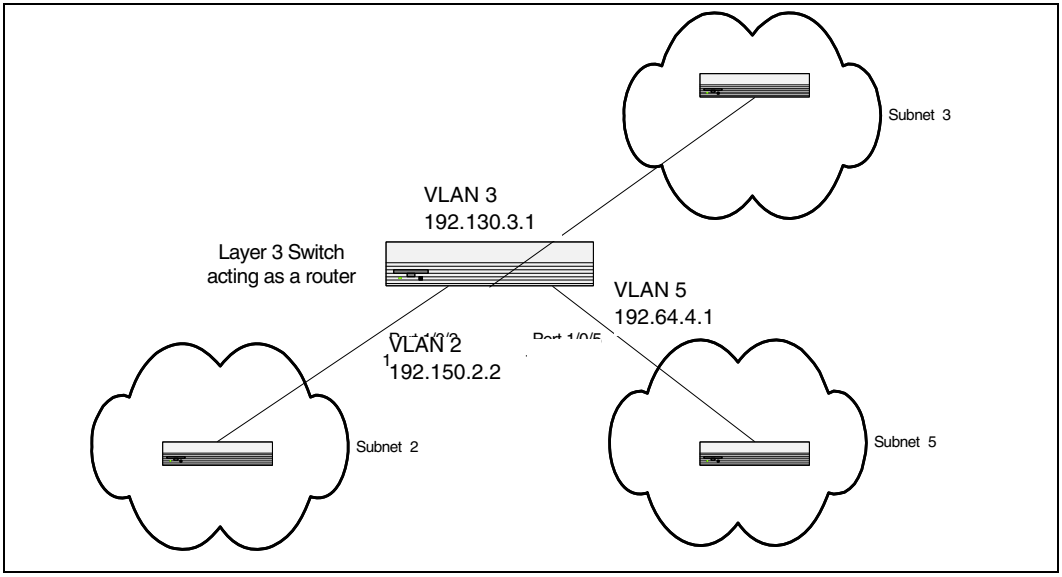


Figure 4-6. Port Routing Example Network Diagram

Example #1: Enable Routing for the Switch

The following sequence enables routing for the switch:

```
console#config
    ip routing
    exit
```

Example #2: Enable Routing for Ports

The following command sequence enables routing and assigns IP addresses for VLAN 2 and VLAN 3.

```
console#config
    interface vlan 2
       routing
       ip address 192.150.2.2 255.255.255.0
       exit
    interface vlan 3
       routing
       ip address 192.130.3.1 255.255.255.0
       exit
    exit
```

Example #3. Enable RIP for the Switch

The next sequence enables RIP for the switch. The route preference defaults to 15.

```
console#config
   router rip
      enable
      exit
   exit
```

Example #4. Enable RIP for the VLAN Routing Interfaces

This command sequence enables RIP for VLAN 2 and VLAN 3. Authentication defaults to none, and no default route entry is created. The commands specify that both interfaces receive both RIP-1 and RIP-2 frames, but send only RIP-2 formatted frames.

```
console#config
   interface vlan 2
      ip rip
      ip rip receive version both
      ip rip send version rip2
      exit
   interface vlan 3
      ip rip
      ip rip receive version both
      ip rip send version rip2
      exit
   exit
```

Using the Web Interface to Configure RIP

Use the following screens to perform the same configuration using the Graphical User Interface:

- **Routing > IP > Configuration>** To enable routing for the switch.
- **Routing > IP > Interface Configuration** > To configure the VLAN routing interfaces.
- **Routing > RIP > Configuration.** To enable RIP for the switch.
- **Routing > RIP > Interface Configuration.** To enable RIP for the VLAN routing interfaces and specify the RIP versions.

Route Preferences

You can use route preference assignment to control how the router chooses which routes to use when alternatives exist. This section describes three uses of route preference assignment:

- "Assigning Administrative Preferences to Routing Protocols" on page 95
- "Using Equal Cost Multipath" on page 97

Assigning Administrative Preferences to Routing Protocols

The router may learn routes from various sources: static configuration, local route discovery, RIP, and OSPF. Most routing protocols use a route metric to determine the shortest path known to the protocol; however, these metrics are independent of one another and not easily comparable. Therefore, when the router learns a route to a particular destination from two different sources, the metrics do not provide a means of choosing the best route for your network.

The PowerConnect 6200 Series switch enables you to identify the preferred route type by assigning an administrative preference value to each type. The values are arbitrary (1 to 255); however, a route type that has a lower value is preferred over higher-value types.

Local routes are assigned an administrative preference value of 0 and are always preferred over other route types to local hosts. Static routes have a default value of 1; however, this value and all other default preference values are user-configurable.

A protocol can be assigned a preference value of 255 to prevent the router from forwarding packets using that protocol.

For routed management traffic:

1. Router entries are checked for applicable destinations.
2. The globally assigned default-gateway is consulted.

Router entries take precedence over an assigned default-gateway.

Example 1: Configure Administrative Preferences

The following commands configure the administrative preference for the RIP and OSPF:

```
console#Config
   router rip
      distance rip 130
      exit
```

For OSPF, an additional parameter identifies the type of OSPF route that the preference value applies to:

```
   router ospf
      distance ospf ?

external                Enter preference value for OSPF external routes.
inter-area              Enter preference value for inter-area routes.
intra-area              Enter preference value for intra-area routes.

      distance ospf inter 170
      exit
```

Example 2: Assigning Administrative Preferences to Static Routes

By default, static routes are assigned a preference value of 1. The following command changes this default:

```
console#Config
   ip route distance 20
   exit
```

When you configure a static route, you can assign a preference value to it. The preference overrides the setting inherited as the default value for static routes.

In this example, two static routes are defined to the same destination but with different next hops and different preferences (25 and 30). The route with the higher preference will only be used when the preferred route is unavailable:

```
console#Config
   ip route 10.25.67.0 255.255.255.0 10.25.22.2
   ip route 10.25.67.0 255.255.255.0 10.25.21.0
   exit
```

Similarly, you can create two default routes—one preferred and the other used as a backup. In this example, the preference values 1 and 10 are assigned:

```
console#Config
   ip route default 10.25.67.2 1
   ip route default 10.25.67.7 10
   exit
```

Using Equal Cost Multipath

The equal cost multipath (ECMP) feature allows a router to use more than one next hop to forward packets to a given destination prefix. It can be used to promote a more optimal use of network resources and bandwidth.

A router that does not use ECMP forwards all packets to a given destination through a single next hop. This next hop may be chosen from among several next hops that provide equally good routes to the destination. For example, in Figure 4-7, Router A sends all traffic to destinations in Network D through next hop NH1, even though the route through NH2 is equally good. Forwarding all traffic via NH1 may cause Link A to be overloaded while Link B is not used at all.

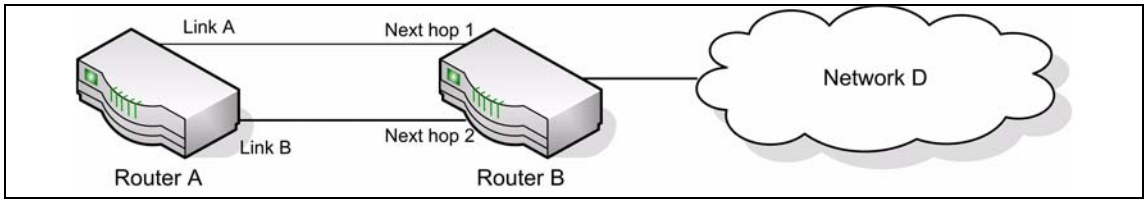


Figure 4-7. Forwarding Without ECMP

With ECMP, Router A can forward traffic to some destinations in Network D via Link A and traffic to other destinations in Network D via Link B, thereby taking advantage of the bandwidth of both links. A hash algorithm is applied to the destination IP addresses to provide a mechanism for selecting among the available ECMP paths.

ECMP routes may be configured statically or learned dynamically. If a user configures multiple static routes to the same destination but with different next hops, then those routes will be treated as a single route with two next hops. For example, given the network in Figure 4-8, if the user configures the following two static routes on Router A, the routing table will contain a single route to 20.0.0.0/8:

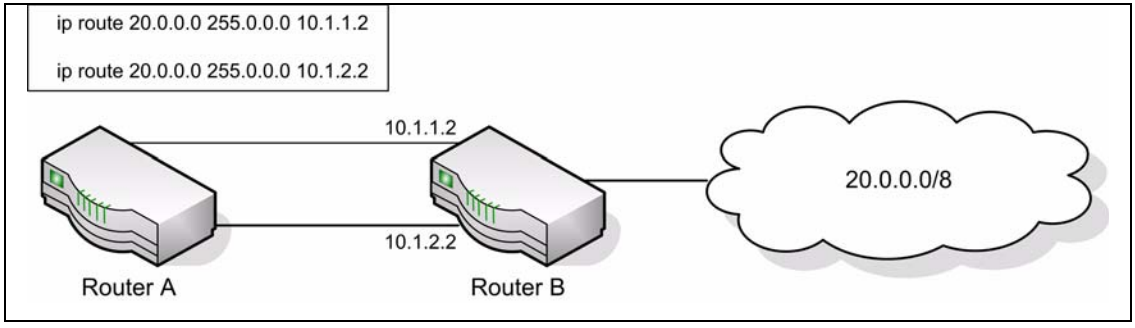


Figure 4-8. Next Hop with Two Static Routes

Routing protocols can also be configured to compute ECMP routes. For example, referring to Figure 4-8, if OSPF were configured in on both links connecting Router A and Router B, and if Router B advertised its connection to 20.0.0.0/8, then Router A could compute an OSPF route to 20.0.0.0/8 with next hops of 10.1.1.2 and 10.1.2.2.

Static and dynamic routes are all included in a single combined routing table. This routing table accepts ECMP routes; however, the routing table will not combine routes from different sources to create ECMP routes. Referring to Figure 4-8, assume OSPF is configured on only one of the links between Router A and Router B. Then, on Router A, assume that OSPF reports to the routing table a route to 20.0.0.0/8 with a next hop of 10.1.1.2. If the user also configures a static route to 20.0.0.0/8 with a single next hop of 10.1.2.2, the routing table will **not** combine the OSPF and static routes into a single route to 20.0.0.0/8 with two next hops. All next hops within an ECMP route must be provided by the same source.

An ECMP route contains only next hops whose paths to the destination are of equal cost. Referring to Figure 4-8, if OSPF were configured on all links, but Router A's interface to the 10.1.1.x network had an OSPF link cost of 5 and its interface to the 10.1.2.x network had an OSPF link cost of 10, then OSPF would use only 10.1.1.2 as the next hop to 20.0.0.0/8.

Example 1: Configuring an ECMP Route

In the following example, two static routes to the same destination are configured to use different next hops (e.g., for load balancing purposes). Note that the preference metric is not specified, so both routes assume the default static route preference of 1.

```
console#Config
    ip route 20.0.0.0 255.0.0.0 10.1.1.2
    ip route 20.0.0.0 255.0.0.0 10.1.2.2
    exit
```

The following command adds a third route with a preference value of 5. This route will be used only when the first two are unreachable:

```
    ip route 20.0.0.0 255.0.0.0 10.1.3.2 5
```

Loopback Interfaces

PowerConnect 6200 Series software provides for the creation, deletion, and management of loopback interfaces.

A loopback interface is a software-only interface that is not associated with a physical location; as such it is not dependent on the physical status of a particular router interface and is always considered "up" as long as the router is running. It enables configuring a stable IP address for remote clients to refer to. The client can communicate with the loopback interface using any available, active router interface.

NOTE: In this context, loopback interfaces should not be confused with the loopback IP address, usually 127.0.0.1, assigned to a host for handling self-routed packets.

Loopbacks are typically used for device management purposes. A client can use the loopback interface to communicate with the router through various services such as telnet and SSH. The address on a loopback behaves identically to any of the local addresses of the router in terms of the processing of incoming packets. This interface provides the source address for sent packets and can receive both local and remote packets.

NOTE: The following example uses the CLI to configure a loopback interface. You can also use the Web interface. Click **Routing > Loopbacks** in the navigation tree.

You can create a loopback interface in the Global Config mode by assigning it a unique ID from 0 to 7:

```
console#configure
console(config)#interface loopback 0
```

Next, you assign an IPv4 or IPv6 address to the interface:

```
console(config-if-loopback0)#ip address 192.168.1.2 255.255.255.255
console(config-if-loopback0)#exit
console(config)#exit
```

You can view the interface configuration from the Privileged Exec mode:

```
console#show ip interface loopback 0

Primary IP Address..............................  192.168.2.2/255.255.255.255
Routing Mode....................................  Enable
Administrative Mode.............................  Enable
Forward Net Directed Broadcasts.................  Disable
Proxy ARP.......................................  Enable
Local Proxy ARP.................................  Disable
Active State....................................  Active
Link Speed Data Rate............................  Inactive
MAC Address.....................................  00FF.F2A3.8888
Encapsulation Type..............................  --------
```

```
IP MTU..................................... 1500
Bandwidth.................................. 100000 kbps
Destination Unreachables................... Enabled
ICMP Redirects............................. Enabled
```

To delete a loopback interface, enter the following command from the Global Config mode:

```
console(config)#no interface loopback 0
console(config)#
```

IP Helper

The IP Helper feature provides the ability for a router to forward configured UDP broadcast packets to a particular IP address. This allows applications to reach servers on non-local subnets. This is possible even when the application is designed to assume a server is always on a local subnet or when the application uses broadcast packets to reach the server (with the limited broadcast address 255.255.255.255, or a network directed broadcast address).

Network administrators can configure relay entries globally and on VLAN routing interfaces. Each relay entry maps an ingress interface and destination UDP port number to a single IPv4 address (the helper address). Multiple relay entries may be configured for the same interface and UDP port, in which case the relay agent relays matching packets to each server address. Interface configuration takes priority over global configuration. If the destination UDP port for a packet matches any entry on the ingress interface, the packet is handled according to the interface configuration. If the packet does not match any entry on the ingress interface, the packet is handled according to the global IP helper configuration.

You can configure discard relay entries. Discard entries are used to discard packets received on a specific interface when those packets would otherwise be relayed according to a global relay entry. Discard relay entries may be configured on VLAN routing interfaces, but are not configured globally.

Additionally, you can configure which UDP ports are forwarded. Certain UDP port numbers can be specified by name in the UI, but network administrators can configure a relay entry with any UDP port number. You can configure relay entries that do not specify a destination UDP port. The relay agent assumes that these entries match packets with the UDP destination ports listed in Table 4-1 (the list of default ports).

Table 4-1. Default Ports - UDP Port Numbers Implied By Wildcard

Protocol	UDP Port Number
IEN-116 Name Service	42
DNS	53
NetBIOS Name Server	137
NetBIOS Datagram Server	138
TACACS Server	49
Time Service	37
DHCP	67
Trivial File Transfer Protocol	69

The switch limits the number of relay entries to four times the maximum number of VLAN routing interfaces (512 relay entries). There is no limit to the number of relay entries on an individual interface, and no limit to the number of servers for a given {interface, UDP port} pair.

NOTE: DHCP relay cannot be enabled and disabled globally. IP helper can be enabled or disabled globally. Enabling IP helper enables DHCP relay.

Certain pre-existing configurable DHCP relay options do not apply to relay of other protocols. These options are unchanged. You can optionally set a maximum hop count or minimum wait time using the **bootpdhcprelay maxhopcount** and **bootpdhcprelay minwaittime** commands.

The relay agent relays DHCP packets in both directions. It relays broadcast packets from the client to one or more DHCP servers, and relays packets to the client that the DHCP server unicasts back to the relay agent. For other protocols, the relay agent only relays broadcast packets from the client to the server. Packets from the server back to the client are assumed to be unicast directly to the client. Because there is no relay in the return direction for protocols other than DHCP, the relay agent retains the source IP address from the original client packet. The relay agent uses a local IP address as the source IP address of relayed DHCP client packets.

When a switch receives a broadcast UDP packet on a routing interface, the relay agent verifies that the interface is configured to relay to the destination UDP port. If so, the relay agent unicasts the packet to the configured server IP addresses. Otherwise, the relay agent verifies that there is a global configuration for the destination UDP port. If so, the relay agent unicasts the packet to the configured server IP addresses. Otherwise the packet is not relayed.

NOTE: If the packet matches a discard relay entry on the ingress interface, the packet is not forwarded, regardless of the global configuration.

The relay agent only relays packets that meet the following conditions:

- The destination MAC address must be the all-ones broadcast address (FF:FF:FF:FF:FF:FF).
- The destination IP address must be the limited broadcast address (255.255.255.255) or a directed broadcast address for the receive interface.
- The IP time-to-live (TTL) must be greater than 1.
- The protocol field in the IP header must be UDP (17).
- The destination UDP port must match a configured relay entry.

CLI Examples

Example 1: Enable/Disable IP Helper

To globally enable/disable IP Helper (relay of UDP packets) use the following command:

```
console (config)#ip helper enable
```

Example 2: Configure IP Helper Globally (DHCP)

To relay DHCP packets received on any interface to two DHCP servers (10.1.1.1 and 10.1.2.1), use the following commands:

```
console (config)#ip helper-address 10.1.1.1 dhcp
console (config)#ip helper-address 10.1.2.1 dhcp
```

Example 3: Enable IP Helper Globally (UDP)

To relay UDP packets received on any interface for all default ports (Table 2) to the server at 20.1.1.1, use the following commands:

```
console (config)#ip helper-address 20.1.1.1
```

Example 4: Enable IP Helper on a VLAN Routing Interface to a Server (DHCP)

To relay DHCP packets received on VLAN 100 to two DHCP servers, 192.168.10.1 and 192.168.20.1, use the following commands:

```
console(config)#interface vlan 100
console(config-if-vlan100)#ip helper-address 192.168.10.1 dhcp
console(config-if-vlan100)#ip helper-address 192.168.20.1 dhcp
```

Example 5: Enable IP Helper on a VLAN Routing Interface to a Server (DHCP and DNS)

To relay DHCP and DNS packets to 192.168.30.1, use the following commands:

```
console(config-if-vlan100)#ip helper-address 192.168.30.1 dhcp
console(config-if-vlan100)#ip helper-address 192.168.30.1 domain
```

Example 6: Enable IP Helper on Multiple VLAN Routing Interfaces

With the following configuration, the relay agent relays:

- DHCP packets received on any interface other than VLAN 200 and VLAN 300 to 192.168.40.1
- DHCP and DNS packets received on VLAN 200 to 192.168.40.2
- SNMP traps (port 162) received on interface VLAN 300 to 192.168.23.1
- Drops DHCP packets received on VLAN 300

```
console(config)#ip helper-address 192.168.40.1 dhcp
console(config)#interface vlan 200
console(config-if-vlan200)#ip helper-address 192.168.40.2 dhcp
console(config-if-vlan200)#ip helper-address 192.168.40.2 domain
console(config-if-vlan200)#exit
console(config)#interface vlan 300
console(config-if-vlan300)#ip helper-address 192.168.23.1 162
console(config-if-vlan300)#ip helper-address discard dhcp
console(config-if-vlan300)#exit
```

Example 7: Show IP Helper Configurations

The following command shows IP Helper configurations:

```
console#show ip helper-a

IP helper is enabled
Interface               UDP Port    Discard  Hit Count     Server Address
------------------- ----------- ---------- ---------- ------------------
vlan 100                  domain         No          0       192.168.30.1
vlan 100                    dhcp         No          0       192.168.10.1
                                                            192.168.20.1
                                                            192.168.30.1
vlan 200                  domain         No          0       192.168.40.2
vlan 200                    dhcp         No          0       192.168.40.2
vlan 300                    dhcp        Yes          0
vlan 300                     162         No          0       192.168.23.1
Any                       Default         No          0       20.1.1.1
Any                         dhcp         No          0       10.1.1.1
                                                            10.1.2.1
```

Example 8: Show IP Helper Statistics

The following command shows IP Helper configurations:

```
console#show ip helper statistics


DHCP client messages received..................  8
DHCP client messages relayed...................  2
DHCP server messages received..................  2
DHCP server messages relayed...................  2
UDP client messages received...................  8
UDP client messages relayed....................  2
DHCP message hop count exceeded max............  0
DHCP message with secs field below min.........  0
DHCP message with giaddr set to local address..  0
Packets with expired TTL.......................  0
Packets that matched a discard entry...........  0
```

5

Device Security

This section describes configuration scenarios for the following features:

- "802.1x Network Access Control" on page 106
- "802.1X Authentication and VLANs" on page 109
- "Authentication Server Filter Assignment" on page 111
- "Access Control Lists (ACLs)" on page 111
- "RADIUS" on page 117
- "TACACS+" on page 120
- "802.1x MAC Authentication Bypass (MAB)" on page 122
- "Captive Portal" on page 125

802.1x Network Access Control

Port-based network access control allows the operation of a system's port(s) to be controlled to ensure that access to its services is permitted only by systems that are authorized to do so.

Port Access Control provides a means of preventing unauthorized access by supplicants or users to the services offered by a system. Control over the access to a switch and the LAN to which it is connected can be desirable in order to restrict access to publicly accessible bridge ports or departmental LANs.

The PowerConnect 6200 Series switch achieves access control by enforcing authentication of supplicants that are attached to an authenticator's controlled ports. The result of the authentication process determines whether the supplicant is authorized to access services on that controlled port.

A PAE (Port Access Entity) can adopt one of two roles within an access control interaction:

- Authenticator – Port that enforces authentication before allowing access to services available via that Port.
- Supplicant – Port that attempts to access services offered by the Authenticator.

Additionally, there exists a third role:

- Authentication server – Server that performs the authentication function necessary to check the credentials of the supplicant on behalf of the Authenticator.

Completion of an authentication exchange requires all three roles. The PowerConnect 6200 Series switch supports the authenticator role only, in which the PAE is responsible for communicating with the supplicant. The authenticator PAE is also responsible for submitting information received from the supplicant to the authentication server in order for the credentials to be checked, which determines the authorization state of the port. Depending on the outcome of the authentication process, the authenticator PAE then controls the authorized/unauthorized state of the controlled Port.

Authentication is accomplished via an external authentication server:

- Remote Authentication Dial-In User Service (RADIUS)
- Terminal Access Controller Access Control System (TACACS+)

802.1x Network Access Control Examples

This section contains examples of the CLI commands used to configure 802.1X.

Example #1: Configure RADIUS Server for Authentication

This example configures a single RADIUS server used for authentication at 10.10.10.10. The shared secret is configured to be *secret*. The process creates a new authentication list, called radiusList, which uses RADIUS as the authentication method. This authentication list is associated with the 802.1x default login. 802.1x port based access control is enabled for the system, and interface 1/g1 is configured to be in force-authorized mode because this is where the RADIUS server and protected network resources are located.

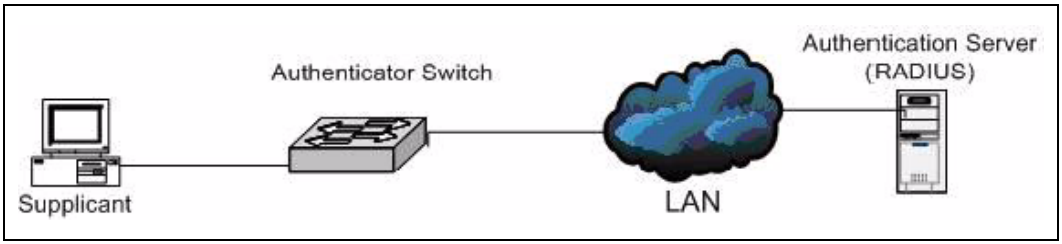


Figure 5-1. Switch with 802.1x Network Access Control

If a user, or supplicant, attempts to communicate via the switch on any interface except interface 1/g1, the system challenges the supplicant for login credentials. The system encrypts the provided information and transmits it to the RADIUS server. If the RADIUS server grants access, the system sets the 802.1x port state of the interface to authorized and the supplicant is able to access network resources.

```
console(config)#radius-server host 10.10.10.10
console(Config-radius)#exit
console(config)#radius-server key secret
console(config)#exit

console#show radius-servers
IP address     Type  Port  TimeOut Retran. DeadTime  Source IP     Prio. Usage
-------------- ----- ----- ------- ------- -------- ------------- ----- -----

10.27.5.157    Auth  1812  Global  Global  Global   10.27.65.13   0     all

Global values
Configured Authentication Servers : 1
Configured Accounting Servers : 0
Named Authentication Server Groups : 1
Named Accounting Server Groups : 0
Timeout : 3
Retransmit : 3
Deadtime : 0
Source IP : 0.0.0.0
RADIUS Attribute 4 Mode : Disable
RADIUS Attribute 4 Value : 0.0.0.0

console(config)#aaa authentication login radiusList radius
console(config)#aaa authentication dot1x default radius
console(config)#dot1x system-auth-control

console(config)#interface ethernet 1/g1
console(config-if-1/g1)#dot1x port-control force-authorized
console(config-if-1/g1)#exit
```

Example #2: MAC-Based Authentication Mode

The PowerConnect 6200 Series switches support MAC-based 802.1X authentication. This feature allows multiple hosts to authenticate on a single port. The hosts are distinguished by their MAC addresses.

When multiple hosts (for example, a PC, a printer, and a phone in the same office) are connected to the switch on the same port, each of the connected hosts authenticates separately with the RADIUS server.

The following command enables MAC-based authentication on port 1/g8 and limits the number of devices that can authenticate on that port to 3. The `switchport mode general` command sets the port to an 802.1Q VLAN. The port must be in general mode in order to enable MAC-based 802.1X authentication.

```
console#configure
console(config)#interface ethernet 1/g8

console(config-if-1/g8)#switchport mode general
console(config-if-1/g8)#dot1x port-control mac-based
console(config-if-1/g8)#dot1x max-users 3
console(config-if-1/g8)#exit
console(config)#exit

console#show dot1x ethernet 1/g8

Administrative Mode.............. Enabled

Port     Admin               Oper          Reauth      Reauth
         Mode                Mode          Control     Period
-------  ------------------  ------------  --------  ----------
1/g8     mac-based           Unauthorized  FALSE     3600

Quiet Period.................................. 60
Transmit Period............................... 30
Maximum Requests.............................. 2
Max Users..................................... 3
Supplicant Timeout............................ 30
Server Timeout (secs)......................... 30


Logical   Supplicant       AuthPAE      Backend   VLAN   Username  Filter
Port      MAC-Address      State        State     Id               Id
-------   --------------   --------     --------  -----  --------  ------
112       0000.0000.0000   Initialize   Idle
```

802.1X Authentication and VLANs

The PowerConnect 6200 Series switches allow a port to be placed into a particular VLAN based on the result of type of 802.1X authentication a client uses when it accesses the switch. The RADIUS server or IEEE 802.1X Authenticator can provide information to the switch about which VLAN to assign the host (supplicant).

When a host connects to a switch that uses a RADIUS server or 802.1X Authenticator to authenticate the host, the host authentication can typically have one of three outcomes:

- The host is authenticated.
- The host attempts to authenticate but fail because it lacks certain security credentials.
- The host is a guest and does not try to authenticate at all.

You can create three separate VLANs on the switch to handle hosts depending on whether the host authenticates, fails the authentication, or is a guest. The RADIUS server informs the switch of the selected VLAN as part of the authentication.

Authenticated and Unauthenticated VLANs

Hosts that authenticate normally use a VLAN that includes access to network resources. Hosts that fail the authentication might be denied access to the network or placed on a "quarantine" VLAN with limited network access.

Much of the configuration to assign hosts to a particular VLAN takes place on the RADIUS server or 802.1X authenticator. If you use an external RADIUS server to manage VLANs, you configure the server to use Tunnel attributes in Access-Accept messages in order to inform the switch about the selected VLAN. These attributes are defined in RFC 2868, and their use for dynamic VLAN is specified in RFC 3580.

The VLAN attributes defined in RFC3580 are as follows:

- Tunnel-Type=VLAN (13)
- Tunnel-Medium-Type=802
- Tunnel-Private-Group-ID=VLANID

VLANID is 12-bits and has a value between 1 and 4093.

Guest VLAN

The Guest VLAN feature allows a switch to provide a distinguished service to unauthenticated users. This feature provides a mechanism to allow visitors and contractors to have network access to reach external network with no ability to browse information on the internal LAN.

In port-based 802.1X mode, when a client that does not support 802.1X is connected to an unauthorized port that is 802.1X-enabled, the client does not respond to the 802.1X requests from the switch. Therefore, the port remains in the unauthorized state, and the client is not granted access to the network. If a guest VLAN is configured for that port, then the port is placed in the configured guest

VLAN and the port is moved to the authorized state, allowing access to the client. However, if the port is in MAC-based 802.1X authentication mode, it will not move to the authorized state. MAC-based mode makes it possible for both authenticated and guest clients to use the same port at the same time.

Client devices that are 802.1X-supplicant-enabled authenticate with the switch when they are plugged into the 802.1X-enabled switch port. The switch verifies the credentials of the client by communicating with an authentication server. If the credentials are verified, the authentication server informs the switch to 'unblock' the switch port and allows the client unrestricted access to the network; i.e., the client is a member of an internal VLAN.

Beginning with software release 2.1, Guest VLAN Supplicant mode is configured on a per-port basis. If a client does not attempt authentication on a port and the port is configured for Guest VLAN, the client is assigned to the guest VLAN configured on that port. The port is assigned a Guest VLAN ID and is moved to the authorized status. Disabling the supplicant mode does not clear the ports that are already authorized and assigned Guest VLAN IDs.

CLI Examples

The following examples show how to configure the switch to accept RADIUS-assigned VLANs and Guest VLANs. The examples assume that the RADIUS server and VLAN information has already been configured on the switch. For information on configuring VLANs, see "Virtual LANs" on page 29.

Example #1: Allow the Switch to Accept RADIUS-Assigned VLANs

The RADIUS server can place a port in a particular VLAN based on the result of the authentication. The command in this example allows the switch to accept VLAN assignment by the RADIUS server.

NOTE: The feature is available in release 2.1 and later.

```
console#config

console(config)#aaa authorization network default radius
```

Example #2: Enable Guest VLANs

This example shows how to set the guest VLAN on interface 1/g20 to VLAN 100. This command automatically enables the Guest VLAN Supplicant Mode on the interface.

NOTE: Define the VLAN before configuring an interface to use it as the guest VLAN.

```
console#configure
console(config)#interface ethernet 1/g20
console(config-if-1/g20)#dot1x guest-vlan 100
console(config-if-1/g20)# <CTRL+Z>

console#show dot1x advanced ethernet 1/g20

Port           Guest       Unauthenticated
               VLAN             Vlan
---------      ---------   ---------------
1/g20           Disabled    Disabled
```

Authentication Server Filter Assignment

The PowerConnect 6200 Series switches allow the external 802.1X Authenticator or RADIUS server to assign DiffServ policies to users that authenticate to the switch. When a host (supplicant) attempts to connect to the network through a port, the switch contacts the 802.1X authenticator or RADIUS server, which then provides information to the switch about which DiffServ policy to assign the host (supplicant). The application of the policy is applied to the host after the authentication process has completed.

To enable filter assignment by an external server, the following conditions must be true:

1. The port that the host is connected to must be enabled for MAC-based port access control by using the following command in Interface Config mode:

   ```
   dot1x port-control mac-based
   ```

2. The RADIUS or 802.1X server must specify the policy to assign.

   For example, if the DiffServ policy to assign is named internet_access, include the following attribute in the RADIUS or 802.1X server configuration:

   Filter-id = “internet_access”

3. The DiffServ policy specified in the attribute must already be configured on the switch, and the policy names must be identical.

   For information about configuring a DiffServ policy, see "Differentiated Services" on page 143. The section, "Example #1: DiffServ Inbound Configuration" on page 144," describes how to configure a policy named internet_access.

   NOTE: If the policy specified within the server attribute does not exist on the switch, authentication will fail.

Access Control Lists (ACLs)

This section describes the Access Control Lists (ACLs) feature.

Overview

Access Control Lists (ACLs) are a collection of permit and deny conditions, called rules, that provide security by blocking unauthorized users and allowing authorized users to access specific resources.

ACLs can also provide traffic flow control, restrict contents of routing updates, and decide which types of traffic are forwarded or blocked. Normally ACLs reside in a firewall router or in a router connecting two internal networks.

The PowerConnect 6200 Series switch supports ACL configuration in both the ingress and egress direction. Egress ACLs provide the capability to implement security rules on the egress flows rather than the ingress flows. Ingress and egress ACLs can be applied to any physical port (including 10G), or port-channel, or VLAN routing port.

Ingress ACLs support Flow-based Mirroring and ACL Logging, which have the following characteristics:

- Flow-based mirroring is the ability to mirror traffic that matches a permit rule to a specific physical port or LAG. Flow-based mirroring is similar to the redirect function, except that in flow-based mirroring a copy of the permitted traffic is delivered to the mirror interface while the packet itself is forwarded normally through the device. You cannot configure a given ACL rule with mirror and redirect attributes.
- ACL Logging provides a means for counting the number of "hits" against an ACL rule. When you configure ACL Logging, you augment the ACL deny rule specification with a "log" parameter that enables hardware hit count collection and reporting. The switch uses a fixed five minute logging interval, at which time trap log entries are written for each ACL logging rule that accumulated a non-zero hit count during that interval. You cannot configure the logging interval.

Using ACLs to mirror traffic is called flow-based mirroring since the traffic flow is defined by the ACL classification rules. This is in contrast to port mirroring, where all traffic encountered on a specific interface is replicated on another interface.

You can set up ACLs to control traffic at Layer 2, Layer 3, or Layer 4. MAC ACLs operate on Layer 2. IP ACLs operate on Layers 3 and 4.

Limitations

The following limitations apply to ingress and egress ACLs.

- Maximum of 100 ACLs.
- Maximum rules per ACL is 127.
- You can configure mirror or redirect attributes for a given ACL rule, but not both.
- The PowerConnect 6200 Series switch supports a limited number of counter resources, so it may not be possible to log every ACL rule. You can define an ACL with any number of logging rules, but the number of rules that are actually logged cannot be determined until the ACL is applied to an interface. Furthermore, hardware counters that become available after an ACL is applied are not retroactively assigned to rules that were unable to be logged (the ACL must be un-applied then re-applied). Rules that are unable to be logged are still active in the ACL for purposes of permitting or denying a matching packet.
- The order of the rules is important: when a packet matches multiple rules, the first rule takes precedence. Also, once you define an ACL for a given port, all traffic not specifically permitted by the ACL is denied access.

NOTE: Although the maximum number of ACLs is 100, and the maximum number of rules per ACL is 127, the system cannot support 100 ACLs that each have 127 rules.

Egress ACL Limitations

Egress ACLs have some additional limitations. The following limitations apply to egress ACLs only:

- Egress ACLs support IP Protocol/Destination, IP Address Source/Destination, L4 Source/Destination port, IP DSCP, IP ToS, and IP precedence match conditions only.
- MAC ACLs are not supported in the egress direction.
- Egress ACLs only support Permit/Deny Action. Logging, mirroring and redirect action are not supported.
- Only one Egress ACL can be applied on an interface. The ACL can have multiple rules to classify flows and apply permit/deny action.
- If the Egress ACLs have "over-lapping" rules, then there can be undesired behavior. This limitation is only applicable if the conflicting ACLs are within the same unit. The restriction is explained below:
  - ACL 1: permit tcp destination port 3000; deny all
  - ACL 2: drop ip source 10.1.1.1; permit all
  - ACL 1 is applied on port 1 and ACL 2 is applied on port 2. Due to this limitation, all the packets egressing port 2 with Source IP 10.1.1.1 and tcp source port 3000 will be permitted even though they should be dropped.

MAC ACLs

MAC ACLs are Layer 2 ACLs. You can configure the rules to inspect the following fields of a packet:

- Source MAC address
- Source MAC mask
- Destination MAC address
- Destination MAC mask
- VLAN ID
- Class of Service (CoS) (802.1p)
- Ethertype

L2 ACLs can apply to one or more interfaces.

Multiple access lists can be applied to a single interface; sequence number determines the order of execution.

You can assign packets to queues using the assign queue option.

IP ACLs

IP ACLs classify for Layers 3 and 4.

Each ACL is a set of up to ten rules applied to inbound traffic. Each rule specifies whether the contents of a given field should be used to permit or deny access to the network, and may apply to one or more of the following fields within a packet:

- Destination IP with wildcard mask
- Destination L4 Port
- Every Packet
- IP DSCP
- IP Precedence
- IP TOS
- Protocol
- Source IP with wildcard mask
- Source L4 port
- Destination Layer 4 port

ACL Configuration Process

To configure ACLs, follow these steps:

1. Create an ACL.
   - Create a MAC ACL by specifying a name.
   - Create an IP ACL by specifying a number.
2. Add new rules to the ACL.
3. Configure the match criteria for the rules.
4. Apply the ACL to one or more interfaces.

IP ACL CLI Example

The script in this section shows you how to set up an IP ACL with two rules, one applicable to TCP traffic and one to UDP traffic. The content of the two rules is the same. TCP and UDP packets will only be accepted by the PowerConnect 6200 Series switch if the source and destination stations have IP addresses that fall within the defined sets.

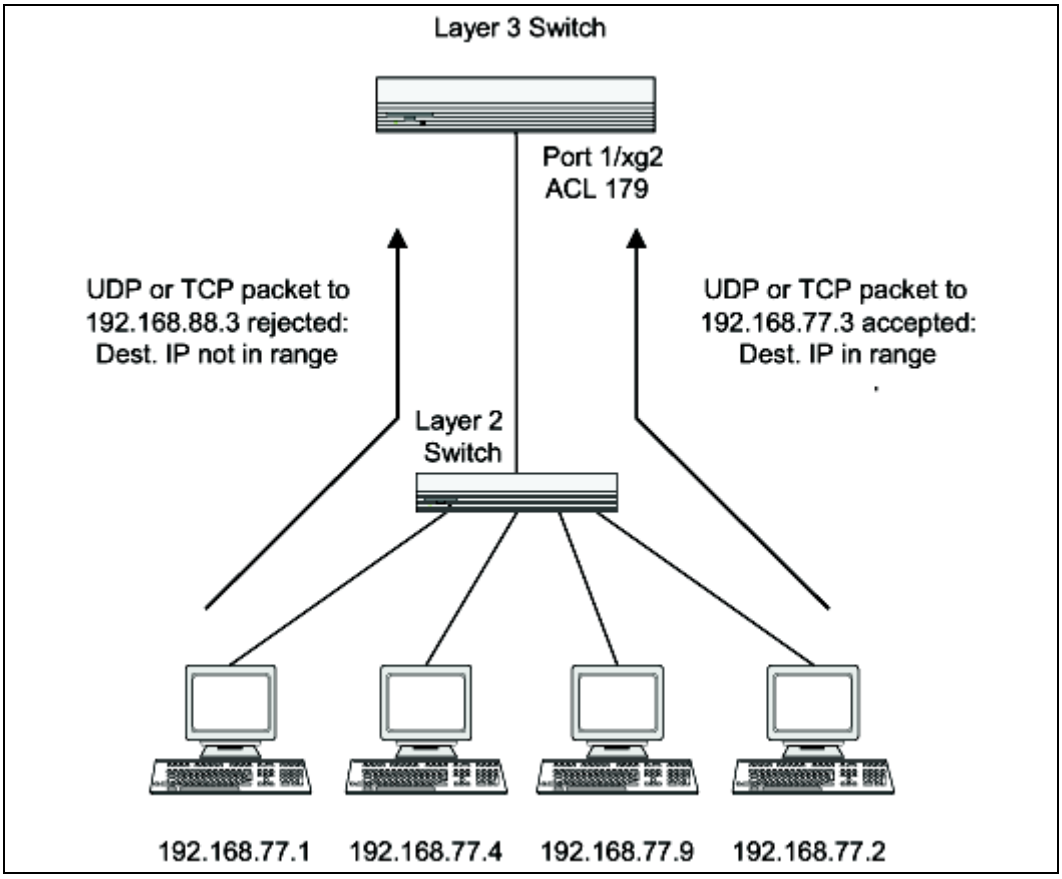


Figure 5-2. IP ACL Example Network Diagram

Step 1: Create an ACL and Define an ACL Rule

This command creates an ACL named list1 and configures a rule for the ACL. After the mask has been applied, it permits packets carrying TCP traffic that matches the specified Source IP address, and sends these packets to the specified Destination IP address.

```
console#config
console(config)#access-list list1 permit tcp 192.168.77.0 0.0.0.255 192.168.77.3
0.0.0.0
```

Step 2: Define the Second Rule for ACL 179

Define the rule to set similar conditions for UDP traffic as for TCP traffic.

```
console(config)#access-list list1 permit udp 192.168.77.0 0.0.0.255 192.168.77.3
0.0.0.255
console(config)#exit
```

Step 3: Apply the Rule to Outbound (Egress) Traffic on Port 1/g2

Only traffic matching the criteria will be accepted.

```
console(config)#interface ethernet 1/g2
console(config-if-1/g2)#ip access-group list1 out
console(config-if-1/g2)#exit
```

Configuring a MAC ACL

The following steps configure a MAC ACL that denies traffic with any MAC address access to hosts with a MAC address of `00:11:22:33:XX:XX`, where `XX` is any hexadecimal value (1-F). The `log` parameter specifies that the system should keep track of the number of times the rule is applied to traffic that meets the rule criteria. When a frame entering the port matches the rule, the rule *hit* counter increments. Every five minutes the ACL application checks the counter. If the counter indicates that the rule has been applied since the last time it was checked, the ACL application logs a message indicating which rule was applied and how many times it was hit during that time period.

The rule is applied to interface 1/g5 in the inbound direction and has a priority value of 6 (the lower the number, the higher the priority).

Step 1: Set up a MAC Access List

```
console#config
console(config)#mac access-list extended mac1
console(config)#exit
```

Step 2: Specify the MAC ACL Attributes

```
console(config-mac-access-list)#deny any 00:11:22:33:44:55 00:00:00:00:FF:FF log
```

Step 3: Configure a MAC Access Group

```
console(config)#interface ethernet 1/g5
console(config-if-1/g5)#mac access-group mac1 in 6
```

Step 4: Viewing the MAC ACL Information

```
console#show mac access-lists

Current number of all ACLs: 2 Maximum number of all ACLs: 100

     MAC ACL Name               Rules       Interface(s)        Direction
------------------------------ ----- ------------------------ ---------
mac1                            1     1/g5                    Inbound

console#show mac access-lists mac1

MAC ACL Name: mac1


Rule Number: 1
Action..................................... deny
Destination MAC Address.................... 00:11:22:33:44:55
Destination MAC Mask....................... 00:00:00:00:FF:FF
Log........................................ TRUE
```

RADIUS

Making use of a single database of accessible information—as in an Authentication Server—can greatly simplify the authentication and management of users in a large network. One such type of Authentication Server supports the Remote Authentication Dial In User Service (RADIUS) protocol as defined by RFC 2865.

For authenticating users prior to access, the RADIUS standard has become the protocol of choice by administrators of large accessible networks. To accomplish the authentication in a secure manner, the RADIUS client and RADIUS server must both be configured with the same shared password or "secret". This "secret" is used to generate one-way encrypted authenticators that are present in all RADIUS packets. The "secret" is never transmitted over the network.

RADIUS conforms to a secure communications client/server model using UDP as a transport protocol. It is extremely flexible, supporting a variety of methods to authenticate and statistically track users. RADIUS is also extensible, allowing for new methods of authentication to be added without disrupting existing functionality.

As a user attempts to connect to a functioning RADIUS supported network, a device referred to as the Network Access Server (NAS) or switch/router first detects the contact. The NAS or user-login interface then prompts the user for a name and password. The NAS encrypts the supplied information and a RADIUS client transports the request to a pre-configured RADIUS server. The server can authenticate the user itself, or make use of a back-end device to ascertain authenticity. In either case a response may or may not be forthcoming to the client. If the server accepts the user, it returns a positive result with

attributes containing configuration information. If the server rejects the user, it returns a negative result. If the server rejects the client or the shared "secrets" differ, the server returns no result. If the server requires additional verification from the user, it returns a challenge, and the request process begins again.

If you use a RADIUS server to authenticate users, you must configure user attributes in the user database on the RADIUS server. The user attributes include the user name, password, and privilege level.

NOTE: To set the privilege level, use the `Service-Type` attribute. Do not us any vendor-specific attribute value pairs.

The following example shows an entry in the FreeRADIUS `/etc/raddb/users` file that allows a user (name: admin) to log onto the switch with read/write privileges, which is equivalent to privilege level 15.

```
admin               Auth-Type := Local,
                    User-Password == "pass1234"
                    Service-Type = NAS-Prompt-User

enable              Auth-Type := Local,
                    User-Password == "pass5678"
                    Service-Type = Administrative-User
```

The values for the Service-Type attribute are as follows:

- `NAS-Prompt-User` indicates the user should be provided a command prompt on the NAS, from which nonprivileged commands can be executed.
- `Administrative-User` indicates the user should be granted access to the administrative interface to the NAS, from which privileged commands can be executed.

RADIUS Configuration Examples

This section contains examples of commands used to configure RADIUS settings on the switch.

Example #1: Basic RADIUS Server Configuration

This example configures two RADIUS servers at 10.10.10.10 and 11.11.11.11. Each server has a unique shared secret key. The shared secrets are configured to be *secret1* and *secret2* respectively. The server at 10.10.10.10 is configured as the primary server. The process creates a new authentication list, called radiusList, which uses RADIUS as the primary authentication method, and local authentication as a backup method in the event that the RADIUS server cannot be contacted.

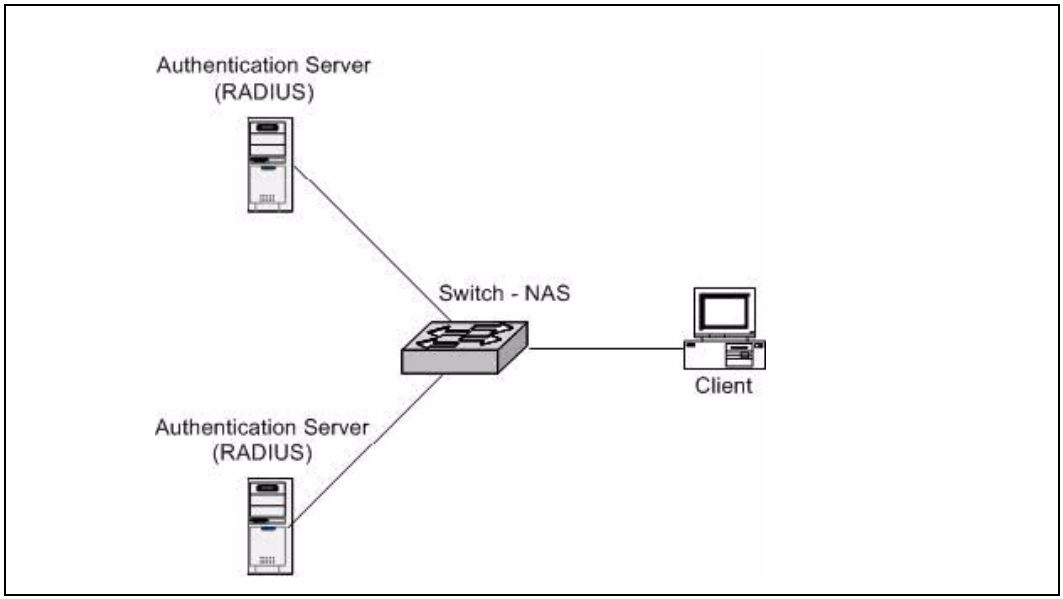


Figure 5-3. RADIUS Servers in a Network

When a user attempts to log in, the switch prompts for a username and password. The switch then attempts to communicate with the primary RADIUS server at 10.10.10.10. Upon successful connection with the server, the login credentials are exchanged over an encrypted channel. The server grants or denies access, which the switch honors, and either allows or does not allow the user to access the switch. If neither of the two servers can be contacted, the switch searches its local user database for the user.

```
console(config)#radius-server host 10.10.10.10
console(Config-radius)#key secret1
console(Config-radius)#priority 1
console(Config-radius)#exit

console(config)#radius-server host 11.11.11.11
console(Config-radius)#key secret2
console(Config-radius)#priority 50
console(Config-radius)#exit

console(config)#aaa authentication login radiusList radius local
console(config)#aaa authentication dot1x default radius
```

Example #2: Set the NAS-IP Address for the RADIUS Server

The NAS-IP address attribute identifies the IP Address of the network authentication server (NAS) that is requesting authentication of the user. The address should be unique to the NAS within the scope of the RADIUS server.

The NAS-IP-Address is only used in Access-Request packets. Either the NAS-IP-Address or NAS-Identifier must be present in an Access-Request packet.

NOTE: The feature is available in release 2.1 and later.

The following command sets the NAS-IP address to 192.168.20.12. If you do not specify an IP address in the command, the NAS-IP address uses the interface IP address that connects the switch to the RADIUS server.

```
console#config
console(config)#radius-server attribute 4 192.168.20.12
```

TACACS+

TACACS+ (Terminal Access Controller Access Control System) provides access control for networked devices via one or more centralized servers. Similar to RADIUS, this protocol simplifies authentication by making use of a single database that can be shared by many clients on a large network. TACACS+ uses TCP to ensure reliable delivery and a shared key configured on the client and daemon server to encrypt all messages.

After you configure TACACS+ as the authentication method for user login, the NAS (Network Access Server) prompts for the user login credentials and requests services from the TACACS+ client. The client then uses the configured list of servers for authentication, and provides results back to the NAS. You can configure the TACACS+ server list with one or more hosts defined via their network IP address. You can also assign each a priority to determine the order in which the TACACS+ client will contact them. TACACS+ contacts the server when a connection attempt fails or times out for a higher priority server.

You can configure each server host with a specific connection type, port, timeout, and shared key, or you can use global configuration for the key and timeout.

Like RADIUS, the TACACS+ server can do the authentication itself, or redirect the request to another back-end device. All sensitive information is encrypted and the shared secret is never passed over the network; it is used only to encrypt the data.

TACACS+ Configuration Example

This example configures two TACACS+ servers at 10.10.10.10 and 11.11.11.11. Each server has a unique shared secret key. The server at 10.10.10.10 has a default priority of 0, the highest priority, while the other server has a priority of 2. The process creates a new authentication list, called tacacsList, which uses TACACS+ to authenticate, and uses local authentication as a backup method.

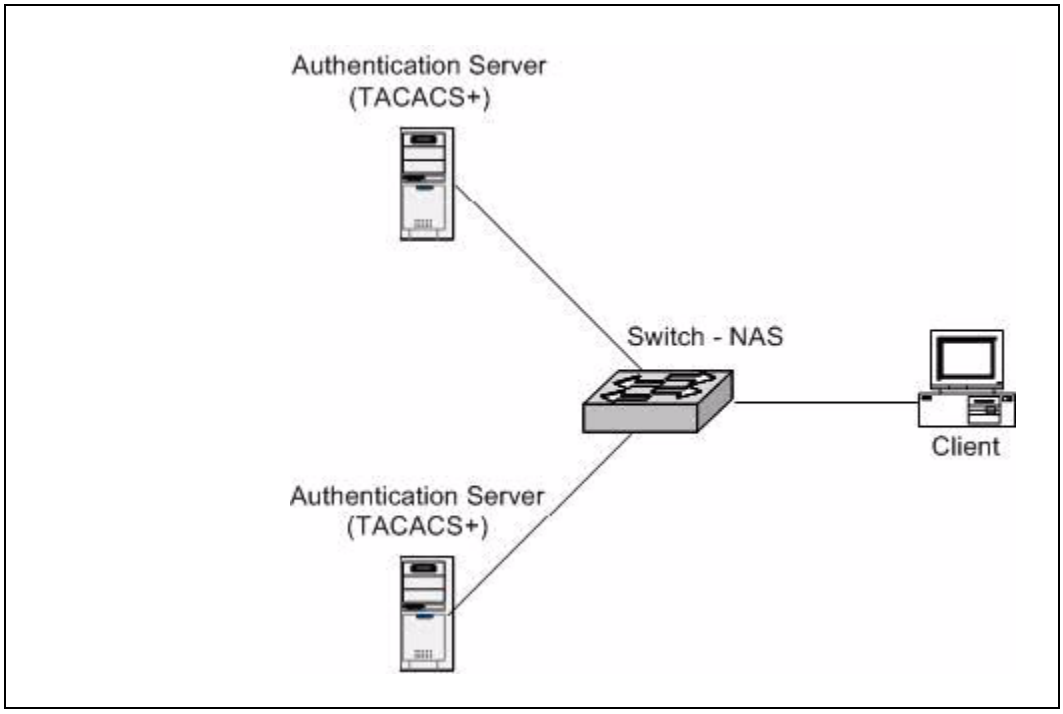


Figure 5-4. PowerConnect 6200 Series Switch with TACACS+

When a user attempts to log into the switch, the NAS or switch prompts for a username and password. The switch attempts to communicate with the highest priority configured TACACS+ server at 10.10.10.10. Upon successful connection with the server, the switch and server exchange the login credentials over an encrypted channel. The server then grants or denies access, which the switch honors, and either allows or does not allow the user to gain access to the switch. If neither of the two servers can be contacted, the switch searches its local user database for the user.

```
console# config
console(config)#tacacs-server host 10.10.10.10
console(config)#key tacacs1
console(config)#exit
console(config)#tacacs-server host 11.11.11.11
console(config)#key tacacs2
console(config)#priority 2
console(config)#exit
console(config)#aaa authentication login tacacsList tacacs local
```

802.1x MAC Authentication Bypass (MAB)

MAB is a supplemental authentication mechanism that allows 802.1x unaware clients, such as printers and fax machines, to authenticate to the network using the client MAC address as an identifier. The known and allowable MAC address and corresponding access rights of the client must be pre-populated in the authentication server. MAB only works when the port control mode of the port is mac-based.

MAB uses the 802.1x infrastructure, and it cannot be supported independent of the Dot1x component.

Operation in the Network

Mac Authentication Bypass (MAB) can be configured on a per–port basis. When a port configured for MAB receives traffic from an unauthenticated client, the switch (Authenticator):

- Sends a EAP Request packet to the unauthenticated client
- Waits a pre-determined period of time for a response
- Retries – resends the EAP Request packet up to three times
- Considers the client to be dot1x unaware client (if it does not receive an EAP response packet from that client)

The authenticator sends a request to the authentication server with the MAC address of the client in 'hhhhhhhhhhhh' format as the username and the MD5 hash of the Mac address as the password. The authentication server checks its database for the authorized Mac addresses and returns an 'Access-Accept' or an 'Access-Reject' (depending on whether the Mac address is found in the database). This also allows dot1x unaware clients to be placed in a RADIUS assigned VLAN or apply a specific Filter ID to the client traffic.

Figure 5-5 illustrates a MAB scenario for:

- No response from the unauthenticated client
- EAPOL timeout
- Access Accept based on MAC address found in database

NOTE: MAB initiates only after the dot1x guest vlan period times out. If the client responds to any of the EAPOL identity requests, MAB does not initiate for that client.

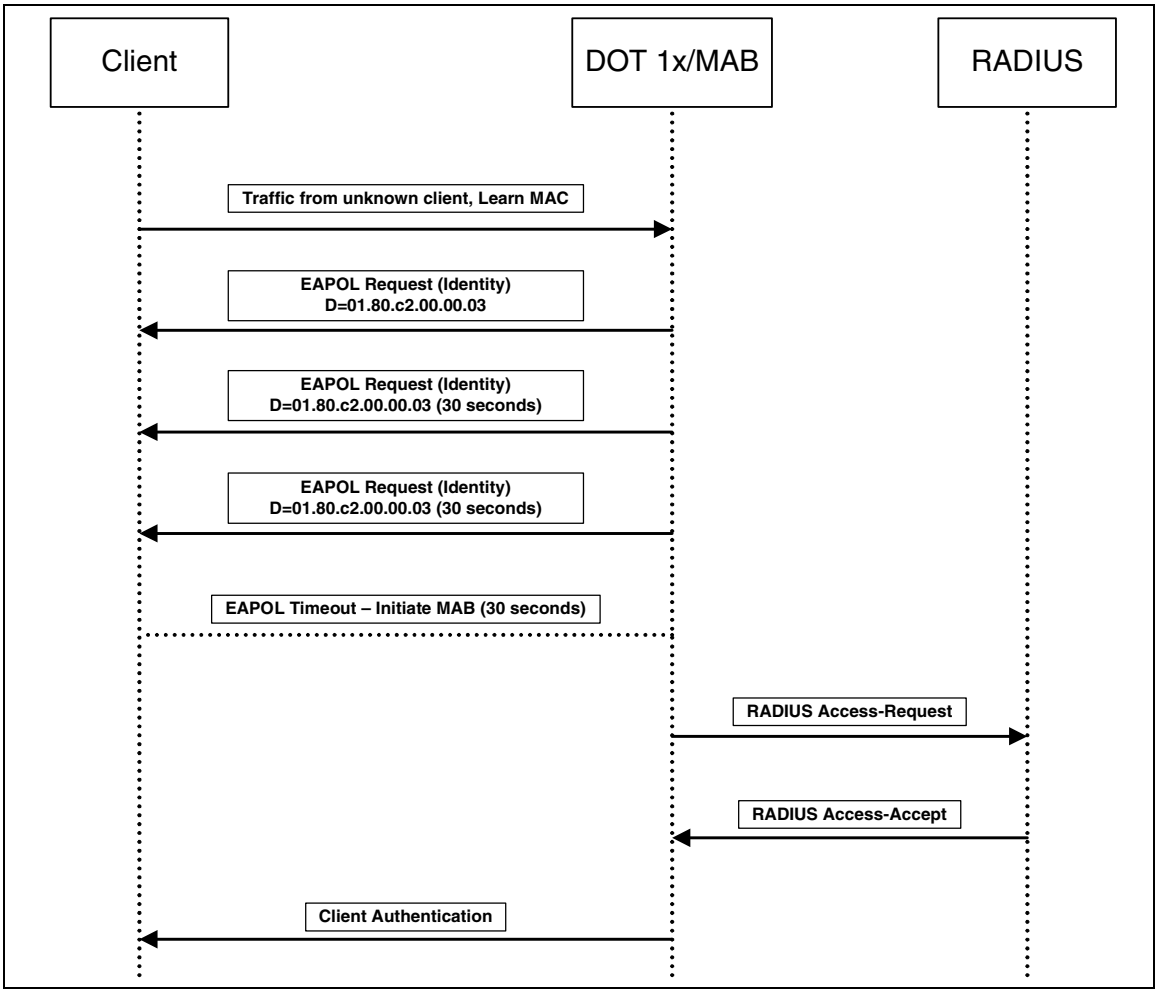


Figure 5-5. MAB Operation – Authentications Based on MAC Address in Database

CLI Examples

Example 1: Enable/Disable MAB

To enable/disable MAB on interface 1/5, use the following commands:

```
console(config-if-1/g5)#dot1x mac-auth-bypass
console(config-if-1/g5)#no dot1x mac-auth-bypass
```

Example 2: Show MAB Configuration

To show the MAB configuration for interface 1/5, use the following command:

```
console#show dot1x ethernet 1/g5

Administrative Mode.............. Enabled

Port     Admin                Oper          Reauth      Reauth
         Mode                 Mode          Control     Period
-------  -------------------  ------------  --------  ----------
1/g5     mac-based            Authorized    TRUE      300

Quiet Period.................................. 60
Transmit Period............................... 30
Maximum Requests.............................. 2
Max Users..................................... 16
Supplicant Timeout............................ 30
Server Timeout (secs)......................... 30
MAB mode (configured)......................... Enabled
MAB mode (operational)........................ Enabled

Logical    Supplicant       AuthPAE        Backend      VLAN  Username
Filter
 Port      MAC-Address      State          State         Id
Id
-------  -----------------  ------------   -----------   ----- --------
--------
   64    0012.43D1.D19F     Authenticated     Idle         1
```

Captive Portal

Overview

Captive Portal feature is a software implementation that allows client access only on user verification. Verification can be configured to allow access for guest and authenticated users. Users must be validated against a database of authorized captive portal users locally or through a radius client.

The Authentication server supports both HTTP and HTTPS web connections. In addition, Captive Portal can be configured to use an optional HTTP or HTTPS port (in support of HTTP Proxy networks). If configured, this additional port is used exclusively by Captive Portal.

NOTE: This optional port is in addition to the default ports (HTTP port 80 and HTTPS port 443), which are used for all other web traffic.

The main captive portal component is a generic implementation that runs within the switch. It provides the network administrator with a common method to configure captive portals for client access. The generic captive portal component handles all configurations, client authentication, and manages status and statistics for presentation to the network administrator communicating with interface-specific components as required.

Functional Description

Captive Portal for wired interfaces allows the clients directly connected to the switch be authenticated using a Captive Portal mechanism before the client is given access to the network.

When a wired physical port is enabled for Captive Portal, the port is set in a captive-portal-enabled state; all traffic coming into the port from unauthenticated clients are dropped except for the ARP, DHCP, DNS, and NETBIOS packets. These packets forwarded by the switch so that the unauthenticated clients can get an IP address resolve the hostname or domain names. Data traffic from authenticated clients is forwarded normally.

All HTTP/HTTPS packets from unauthenticated clients are directed to the CPU on the switch for the ports that are enabled for Captive Portal. When an unauthenticated client opens a web browser and tries to connect to network, the Captive Portal redirects all the HTTP/HTTPS traffic from unauthenticated clients to the authenticating server on the switch. A Captive portal web page is sent back to the unauthenticated client and the client can authenticate and gain access to the port.

The Captive Portal feature can be enabled on all physical ports on the switch. It is not supported for VLAN interfaces, loopback interfaces, or logical interfaces.

The Captive Portal feature performs Mac-based authentication (not port-based authentication). All clients connected to the captive portal interface must be authenticated before accessing the network.

There are three states for clients connecting to the Captive Portal interface:

- Unknown State
- Unauthenticated State
- Authenticated State

In the unknown state, the CP doesn't redirect HTTP/S traffic to the switch, but queries the switch to determine whether the client is authenticated or unauthenticated.

In the Unauthenticated state, the CP directs the HTTP/S traffic to the switch to allow the client to authenticate with the switch.

Once the client is authenticated, the client is placed in Authenticated state; in this state all the traffic emerging from the client will be forwarded through the switch.

Captive Portal Configuration, Status and Statistics

This section describes the configurations, status, and statistics that can be viewed by a network administrator.

Captive Portal customized web pages are only configurable via the Web Interface. Otherwise, the configurations included in this section are managed using the standard management interfaces (Web, CLI, and SNMP).

Captive Portal Configuration

The Captive Portal configuration allows the network administrator to control:

- Verification and authentication
- Assignment to interfaces
- Client sessions
- Web page customization

The administrator can create up to 10 captive portal configuration instances. Each configuration contains flags and definitions for controlling client access, and content used to customize the user verification web page. A captive portal configuration can be applied to one or more interfaces. An interface may only be a physical port on the switch.

Client Access, Authentication, and Control

User verification can be configured to allow access for guest users; users that do not have assigned user names and passwords. User verification can also be configured to allow access for authenticated users. Authenticated users are required to enter a valid user name and password that are validated against the local database or a RADIUS server. Network access is granted once user verification has been confirmed.

The administrator can block access to a captive portal configuration. When an instance is blocked, no client traffic is allowed through any associated interfaces. Blocking a captive portal instance is a temporary command executed by the administrator (not saved in the configuration).

When using Local authentication, the administrator provides user identities for Captive Portal by adding unique user names and passwords to the Local User Database.

This configuration is global to the captive portal component and can contain up to 128 user entries (a RADIUS server should be used if more users are required). A local user can belong to only one group. There is one group created by default with the group name "Default" to which all new users are assigned.

All new captive portal instances are also assigned to the "Default" group. The administrator can create new groups and modify the user/group association to only allow a subset of users access to a specific captive portal instance. Network access is granted upon successful verification of user credentials.

A remote RADIUS server can be used for client authentication. RADIUS authentication and accounting servers are configured separately from the captive portal configuration. In order to perform authentication/accounting via RADIUS, the administrator configures one or more RADIUS servers and then references the server(s) using their name in the captive portal configuration (each captive portal instance can be assigned one RADIUS authentication server and one RADIUS accounting server). If RADIUS is enabled for a captive portal configuration and no RADIUS servers are assigned, the captive portal activation status indicates the instance is disabled with an appropriate reason code.

The Table 5-1 shows the RADIUS attributes that are used to configure captive portal users. The table indicates both RADIUS attributes and vendor specific attributes (VSA) that are used to configure Captive Portal. VSAs are denoted in the id column and are comma delimited (vendor id, attribute id).

Table 5-1. Captive Portal RADIUS Attributes

Radius Attribute	#	Description	Range	Usage	Default
User-Name	1	User name to be authorized	1-32 characters	Required	None
User-Password	2	User password	8-64 characters	Required	None
Session-Timeout	27	Logout once session timeout is reached (seconds). If the attribute is 0 or not present then use the value configured for the captive portal.	Integer (seconds)	Optional	0
Captive-Portal-Groups	6231, 127	A comma-delimited list of group names that correspond to the configured CP instance configurations.	String	Optional	None; the default group is used if not defined here.

A Captive Portal instance can be configured to use the HTTPS protocol during its user verification process. The connection method for HTTPS uses the Secure Sockets Layer (SSL) protocol which requires a certificate to provide encryption. The certificate is presented to the user at connection time.

The Captive Portal component uses the same certificate that is used by the switch for Secure HTTP connections. This certificate can be generated by the administrator using a CLI command. If a captive portal instance is configured for the HTTPS protocol and there is not a valid certificate present on the system, the captive portal instance status shows Disabled with an appropriate reason code.

Client Authentication Logout Request

The administrator can configure and enable 'user logout'. This feature allows the authenticated client to deauthenticate from the network.

In response to the request, the authenticated user is removed from the connection status tables. If the client logout request feature is not enabled, or the user does not specifically request logout, the connection status remains authenticated until Captive Portal deauthenticates (session timeout, idle time, etc.). In order for user logout to function properly, the client browser must be configured such that JavaScript is enabled and popup windows are allowed.

Web Page Customization

Captive Portal provides a web interface to create and customize a specific web page for each Captive Portal configuration. This is accomplished by providing text input components that accept Unicode literal characters.

NOTE: Customization of Captive Portal web pages is accomplished using the Web UI and is not available by using the CLI.

Figure 5-6. PowerConnect 6200 Series Switch with TACACS+

The administrator can download and configure image files for branding purposes. Each image must first be copied onto the switch; Captive Portal provides a HTTP file browser component for this purpose. GIF (Graphics Interchange Format) and/or JPEG (Joint Photographic Experts Group) file types are supported. Once an image file is copied to the switch it can be selected from a drop-down list and associated with a specific web page configuration.

The authentication server generates user verification pages upon receipt of a specific URL request. The URL provides an interface identifier that links to the data in the Captive Portal configuration. The authentication server reads the associated data to construct and serve the appropriate web page.

Captive Portal Status

Captive Portal status is available primarily through 3 tables:

- Client Connections
- Authentication Failures
- Activity Log

Client Connections

Client entries are added to and deleted from this table as each user becomes authenticated or de-authenticated using Captive Portal. A trap is sent for every addition. Each table entry identifies the authenticated user, the connection interface, and the captive portal instance for which the client is authenticated and the current session time.

The administrator may issue a command to de-authenticate a connected client. As a result, the client session is terminated and the associated entry is removed from the database. This does not prevent the user from obtaining a subsequent captive portal connection. The administrator must remove the user entry from the local user database (or RADIUS) configuration to prevent future connections.

The size of the table has a limit of 1024 entries. If the list becomes full, new table entries are rejected and a trap is sent for every rejected client.

Captive Portal Statistics

Client session statistics are available for both guest and authenticated users.Client statistics are used to enforce the idle timeout and other limits configured for the user and captive portal instance. Client statistics may not be cleared by the administrator since this would affect the ability to monitor the configured limits.

CLI Examples

Example 1: Enter Captive Portal configuration mode

To enter Captive Portal configuration mode, use the following command:

```
console(config)#captive-portal
console(config-CP)#
```

Example 2: Enable Captive Portal

To globally enable Captive Portal, use the following command (Captive Portal configuration mode):

```
console(config-CP)#enable
```

Example 3: Enable Captive Portal on Additional HTTP Port

To configure an additional HTTP port for Captive Portal to monitor, use the following command (Captive Portal configuration mode):

```
console(config-CP)#http port 81
```

Example 4: Configure Captive Portal Authentication Timeout

To configure the Captive Portal authentication timeout (600 seconds), use the following command (Captive Portal configuration mode):

```
console(config-CP)#authentication timeout 600
```

Example 5: Show Captive Portal

To show the status of Captive Portal, use the following command:

```
console#show captive-portal
Administrative Mode..................... Enabled
Operational Status...................... Enabled
Disable Reason.......................... Administrator Disabled
Captive Portal IP Address............... 1.2.3.4
```

Example 6: Show Captive Portal Instances

To show the status of all Captive Portal instances in the system, use the following command:

```
console#show captive-portal status

Additional HTTP Port.......................... 81
Additional HTTP Secure Port................... 0
Peer Switch Statistics Reporting Interval..... 300
Authentication Timeout........................ 600
Supported Captive Portals..................... 10
Configured Captive Portals.................... 2
Active Captive Portals........................ 1
System Supported Users........................ 1024
Local Supported Users......................... 128
Authenticated Users........................... 0
```

Example 7: Modify the Default Captive Portal Configuration (Change Verification Method to Local)

To change the verification method to local, use the following command:

```
console(config-CP 1)#verification local
```

To view the configuration change, use the following command:

```
console#show captive-portal configuration 1 status

CP ID..................................... 1
CP Name................................... Default
CP Mode................................... Enable
Protocol Mode............................. HTTP
Verification Mode......................... Local
Group ID.................................. 1
Group Name................................ Default
User Logout Mode.......................... Enable
URL Redirect Mode......................... Disable
Session Timeout........................... 0
Idle Timeout.............................. 0
Max Bandwidth Up (bytes/sec).............. 0
Max Bandwidth Down (bytes/sec)............ 0
Max Input Octets (bytes).................. 0
Max Output Octets (bytes)................. 0
Max Total Octets (bytes).................. 0
```

To create a local user, use the following command:

```
console(Config-CP)#user 1 name user1
console(config-CP)#user 1 password
Enter password (8 to 64 characters): ********
Re-enter password: ********
console(Config-CP)#user 1 session-timeout 14400
```

To verify the creation of a local user, use the following command:

```
console#show captive-portal user
                              Session  Idle
User ID       User Name       Timeout Timeout  Group ID      Group Name
------- --------------------- ------- -------- -------- ---------------------
1       user1                 14400   0        1        Default
```

Example 8: Associate an Interface with a Captive Portal Configuration

To associate an interface with a Captive Portal configuration, use the following command:

```
console#configure
(Config)#captive-portal
(Config-CP)#configuration 1
console(Config-CP 1)#interface 1/g18
```

To view the new interface, use the following command:

```
console#show captive-portal configuration 1 interface

CP ID..................................... 1
CP Name................................... Default
```

```
                                                    Operational    Block
Interface        Interface Description                 Status      Status
--------- ---------------------------------------- ------------ -----------
1/g18     Unit: 1 Slot: 0 Port: 18 Gigabit - Level Disabled     Not Blocked
```

To view the status of a captive client (connected to 1/g18), use the following command:

```
console#show captive-portal configuration 1 client status

CP ID..................................... 1
CP Name................................... Default

     Client           Client
   MAC Address      IP Address    Interface        Interface Description
---------------- --------------- --------- ----------------------------------
00:12:79:BF:94:7A 192.168.1.10    1/g18     Slot: 1 Port: 18 Gigabit - Level

This command shows a statistics for the above client
#show captive-portal client 00:12:79:BF:94:7A statistics

Client MAC Address.......................... 00:12:79:BF:94:7A
Bytes Received.............................. 10541
Bytes Transmitted........................... 47447
Packets Received............................ 78
Packets Transmitted......................... 71
```

6

IPv6

This section includes the following subsections:

- "Overview" on page 135
- "Interface Configuration" on page 135

Overview

There are many conceptual similarities between IPv4 and IPv6 network operation. Addresses still have a network prefix portion (subnet) and a device interface specific portion (host). While the length of the network portion is still variable, most users have standardized on using a network prefix length of 64 bits. This leaves 64 bits for the interface specific portion, called an Interface ID in IPv6. Depending upon the underlying link addressing, the Interface ID can be automatically computed from the link (e.g., MAC address). Such an automatically computed Interface ID is called an EUI64 identifier.

IPv6 packets on the network are of an entirely different format than traditional IPv4 packets and are also encapsulated in a different EtherType (contained within the L2 header to indicate which L3 protocol is used). In order to route these packets across L3 requires an infrastructure equivalent to and parallel to that provided for IPv4.

NOTE: The PowerConnect 6200 Series switch also implements OSPFv3 for use with IPv6 networks. These configuration scenarios are included with the OSPFv2 scenarios in "OSPF" on page 81.

Interface Configuration

In PowerConnect 6200 Series software, IPv6 coexists with IPv4. As with IPv4, IPv6 routing can be enabled on physical and VLAN interfaces. Each L3 routing interface can be used for IPv4, IPv6, or both.

Neighbor discovery is the IPv6 replacement for Address Resolution Protocol (ARP). Router advertisement is part of the neighbor discovery process and is required for IPv6. As part of router advertisement, PowerConnect 6200 Series software supports stateless auto configuration of end nodes. The switch supports both EUI-64 interface identifiers and manually configured interface IDs.

While optional in IPv4, router advertisement is mandatory in IPv6. Router advertisements specify the network prefix(es) on a link which can be used by receiving hosts, in conjunction with an EUI64 identifier, to auto configure a host's address. Routers have their network prefixes configured and may use EUI64 or manually configured interface IDs. In addition to one or more global addresses, each IPv6 interface also has an auto-configured link-local address which is:

- Allocated from part of the IPv6 unicast address space
- Not visible off the local link
- Not globally unique

Next hop addresses computed by routing protocols are usually link-local.

During a transition period, a global IPv6 Internet backbone may not be available. The solution of this is to tunnel IPv6 packets inside IPv4 to reach remote IPv6 islands. When a packet is sent over such a link, it is encapsulated in IPv4 in order to traverse an IPv4 network and has the IPv4 headers removed at the other end of the tunnel.

CLI Example

In Figure 6-1, two devices are connected as shown in the diagram. The VLAN 15 routing interface on both devices connects to an IPv4 backbone network where OSPF is used as the dynamic routing protocol to exchange IPv4 routes. OSPF allows device 1 and device 2 to learn routes to each other (from the 20.20.20.x network to the 10.10.10.x network and vice versa). The VLAN 2 routing interface on both devices connects to the local IPv6 network. OSPFv3 is used to exchange IPv6 routes between the two devices. The tunnel interface allows data to be transported between the two remote IPv6 networks over the IPv4 network.

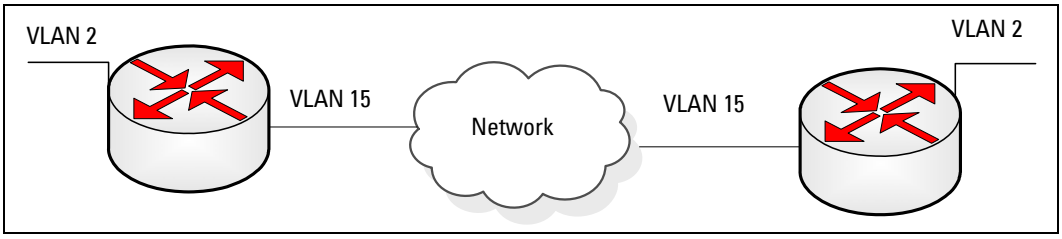


Figure 6-1. IPv6 Example

Device 1

```
console# config
    ip routing
    ipv6 unicast-routing
    router ospf
       router-id 1.1.1.1
       exit

    ipv6 router ospf
       router-id 1.1.1.1
       exit

    interface vlan 15
       routing
       ip address 20.20.20.1 255.255.255.0
```

```
    ip ospf area 0.0.0.0
    exit

interface vlan 2
    routing
    ipv6 enable
    ipv6 address 2020:1::1/64
    ipv6 ospf
    ipv6 ospf network point-to-point
    exit

interface tunnel 0
    ipv6 address 2001::1/64
    tunnel mode ipv6ip
    tunnel source 20.20.20.1
    tunnel destination 10.10.10.1
    ipv6 ospf
    ipv6 ospf network point-to-point
    exit

interface loopback 0
    ip address 1.1.1.1 255.255.255.0
    exit
exit
```

Device 2

```
console# config
    ip routing
    ipv6 unicast-routing
    router ospf
        router-id 2.2.2.2
        exit

    ipv6 router ospf
        router-id 2.2.2.2
        exit

    interface vlan 15
        routing
        ip address 10.10.10.1 255.255.255.0
        ip ospf area 0.0.0.0
        exit

    interface vlan 2
        routing
        ipv6 enable
```

```
    ipv6 address 2020:2::2/64
    ipv6 ospf
    ipv6 ospf network point-to-point
    exit

interface tunnel 0
    ipv6 address 2001::2/64
    tunnel mode ipv6ip
    tunnel source 10.10.10.1
    tunnel destination 20.20.20.1
    ipv6 ospf
    ipv6 ospf network point-to-point
    exit

interface loopback 0
    ip address 2.2.2.2 255.255.255.0
    exit
exit
```

7

Quality of Service

This section includes the following subsections:

- "Class of Service Queuing" on page 139
- "Differentiated Services" on page 143

Class of Service Queuing

The Class of Service (CoS) feature lets you give preferential treatment to certain types of traffic over others. To set up this preferential treatment, you can configure the ingress ports, the egress ports, and individual queues on the egress ports to provide customization that suits your environment.

The level of service is determined by the egress port queue to which the traffic is assigned. When traffic is queued for transmission, the rate at which it is serviced depends on how the queue is configured and possibly the amount of traffic present in other queues for that port.

Some traffic is classified for service (i.e., packet marking) before it arrives at the switch. If you decide to use these classifications, you can map this traffic to egress queues by setting up a CoS Mapping table.

Each ingress port on the switch has a default priority value (set by configuring VLAN Port Priority in the Switching sub-menu) that determines the egress queue its traffic gets forwarded to. Packets that arrive without a priority designation, or packets from ports you've identified as "untrusted," get forwarded according to this default.

Ingress Port Configuration

Trusted and Untrusted Ports/CoS Mapping Table

The first task for ingress port configuration is to specify whether traffic arriving on a given port is "trusted" or "untrusted."

A trusted port means that the system will accept at face value a priority designation within arriving packets. You can configure the system to trust priority designations based on one of the following fields in the packet header:

- 802.1 Priority: values 0-7
- IP DSCP: values 0-63

You can also configure an ingress port as untrusted, where the system ignores priority designations of incoming packets and sends the packet to a queue based on the ingress port's default priority.

CoS Mapping Table for Trusted Ports

Mapping is from the designated field values on trusted ports' incoming packets to a traffic class priority (actually a CoS traffic queue). The trusted port field-to-traffic class configuration entries form the Mapping Table the switch uses to direct ingress packets from trusted ports to egress queues.

Egress Port Configuration—Traffic Shaping

For unit/slot/port interfaces, you can specify the shaping rate for the port (in Kbps), which is an upper limit of the transmission bandwidth used.

Queue configuration

For each queue, you can specify:

- Minimum bandwidth guarantee
- Scheduler type – strict/weighted: Strict priority scheduling gives an absolute priority, with highest priority queues always sent first, and lowest priority queues always sent last. Weighted scheduling requires a specification of priority for each queue relative to the other queues, based on their minimum bandwidth values.

Queue Management Type

The switch supports the tail drop method of queue management. This means that any packet forwarded to a full queue is dropped regardless of its importance.

CLI Examples

Figure 7-1 illustrates the network operation as it relates to CoS mapping and queue configuration.

Four packets arrive at the ingress port 1/g10 in the order A, B, C, and D. You've configured port 1/g10 to trust the 802.1p field of the packet, which serves to direct packets A, B, and D to their respective queues on the egress port. These three packets utilize port 1/g10's 802.1p to COS Mapping Table. In this case, the 802.1p user priority 3 was set up to send the packet to queue 5 instead of the default queue 3. Since packet C does not contain a VLAN tag, the 802.1p user priority does not exist, so Port 1/g10 relies on its default port priority (2) to direct packet C to egress queue 1.

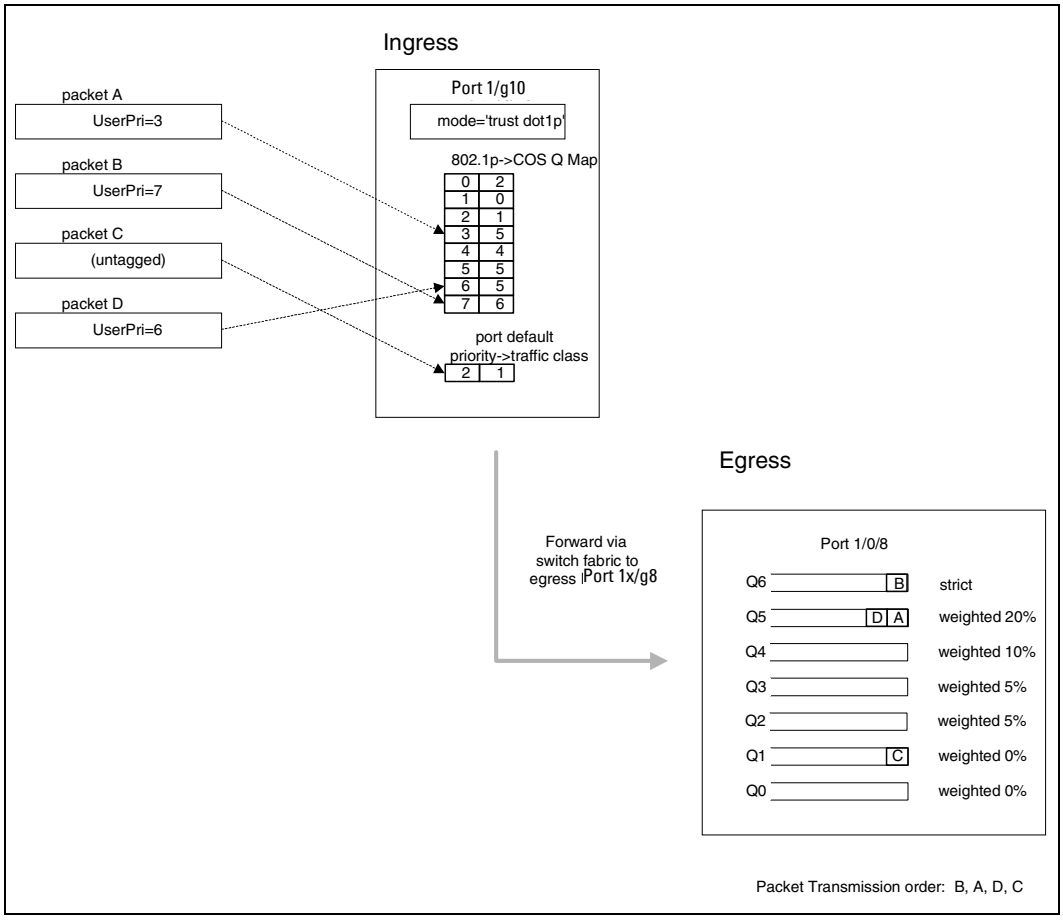


Figure 7-1. CoS Mapping and Queue Configuration

Continuing this example, you configured the egress Port 1/g8 for strict priority on queue 6, and a set a weighted scheduling scheme for queues 5-0. Assuming queue 5 has a higher weighting than queue 1 (relative weight values shown as a percentage, with 0% indicating the bandwidth is not guaranteed), the queue service order is 6 followed by 5 followed by 1. Assuming each queue unloads all packets shown in the diagram, the packet transmission order as seen on the network leading out of Port 1/g8 is B, A, D, C. Thus, packet B, with its higher user precedence than the others, is able to work its way through the device with minimal delay and is transmitted ahead of the other packets at the egress port.

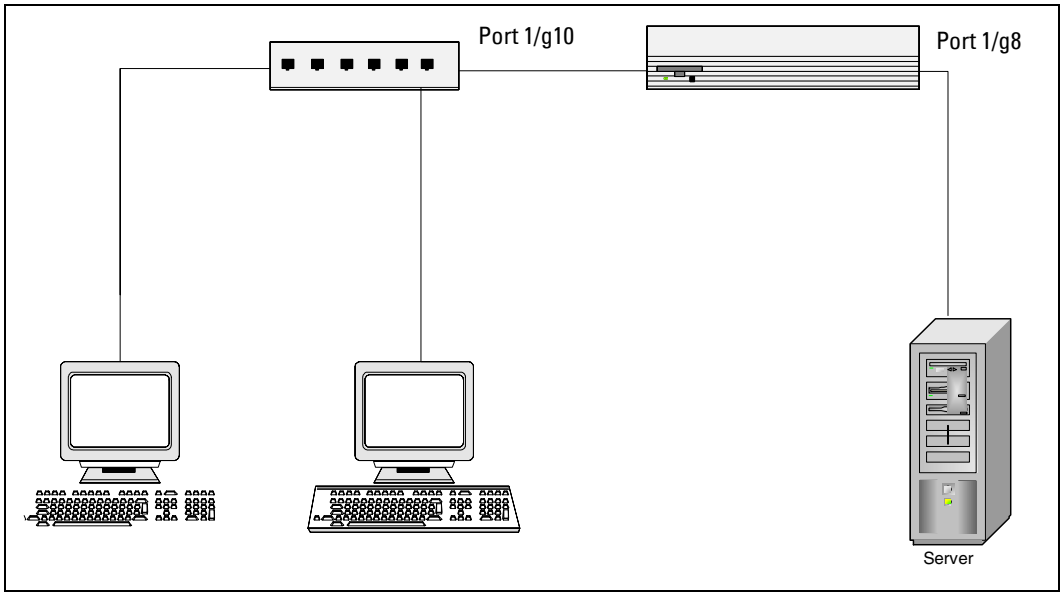


Figure 7-2. CoS1/g Configuration Example System Diagram

You will configure the ingress interface uniquely for all cos-queue and VLAN parameters.

```
console#config
   interface ethernet 1/g10
      classofservice trust dot1p
      classofservice dot1p-mapping 6 3
      vlan priority 2
      exit

   interface ethernet 1/g8
      cos-queue min-bandwidth 0 0 5 5 10 20 40
      cos-queue strict 6
      exit
   exit
```

You can also set traffic shaping parameters for the interface. If you wish to shape the egress interface for a sustained maximum data rate of 80 Kbps (assuming a 100Mbps link speed), you would add a simple configuration line expressing the shaping rate as a percentage of link speed.

```
console#config
   interface ethernet 1/g8
      traffic-shape 42200 kbps
      exit
   exit
```

Differentiated Services

Differentiated Services (DiffServ) is one technique for implementing Quality of Service (QoS) policies. Using DiffServ in your network allows you to directly configure the relevant parameters on the switches and routers rather than using a resource reservation protocol.This section explains how to configure the switch to identify which traffic class a packet belongs to, and how it should be handled to provide the desired quality of service. As implemented in PowerConnect 6200 Series software, DiffServ allows you to control what traffic is accepted and what traffic is discarded.

Traffic to be processed by the DiffServ feature requires an IP header if the system uses IP Precedence or IP DSCP marking.

How you configure DiffServ support in PowerConnect 6200 Series software varies depending on the role of the switch in your network:

- **Edge device**: An edge device handles ingress traffic, flowing towards the core of the network, and egress traffic, flowing away from the core. An edge device segregates inbound traffic into a small set of traffic classes, and is responsible for determining a packet's classification. Classification is primarily based on the contents of the Layer 3 and Layer 4 headers, and is recorded in the Differentiated Services Code Point (DSCP) added to a packet's IP header.
- **Interior node**: A switch in the core of the network is responsible for forwarding packets, rather than for classifying them. It decodes the DSCP in an incoming packet, and provides buffering and forwarding services using the appropriate queue management algorithms.

Before configuring DiffServ on a particular PowerConnect 6200 Series switch, you must determine the QoS requirements for the network as a whole in terms of rules, which are used to classify inbound traffic on a particular interface. The switch does not support DiffServ in the outbound direction.

During configuration, you define DiffServ rules in terms of classes, policies and services:

- **Class**: A class consists of a set of rules that identify which packets belong to the class. Inbound traffic is separated into traffic classes based on Layer 2, Layer 3, and Layer 4 header data. One class type is supported, **All,** which specifies that every match criterion defined for the class must be true for a match to occur.
- **Policy**: Defines the QoS attributes for one or more traffic classes. An example of an attribute is the ability to mark a packet at ingress. The switch supports the ability to assign traffic classes to output CoS queues, and to mirror incoming packets in a traffic stream to a specific egress interface (physical port or LAG).
  PowerConnect 6200 Series software supports the **Traffic Conditioning Policy** type which is associated with an inbound traffic class and specifies the actions to be performed on packets meeting the class rules:
  - Marking the packet with a given DSCP, IP precedence, or CoS
  - Policing packets by dropping or re-marking those that exceed the class's assigned data rate
  - Counting the traffic within the class
- **Service –** Assigns a policy to an interface for inbound traffic.

CLI Example

This example shows how a network administrator can provide equal access to the Internet (or other external network) to different departments within a company. Each of four departments has its own Class B subnet that is allocated 25% of the available bandwidth on the port accessing the Internet.

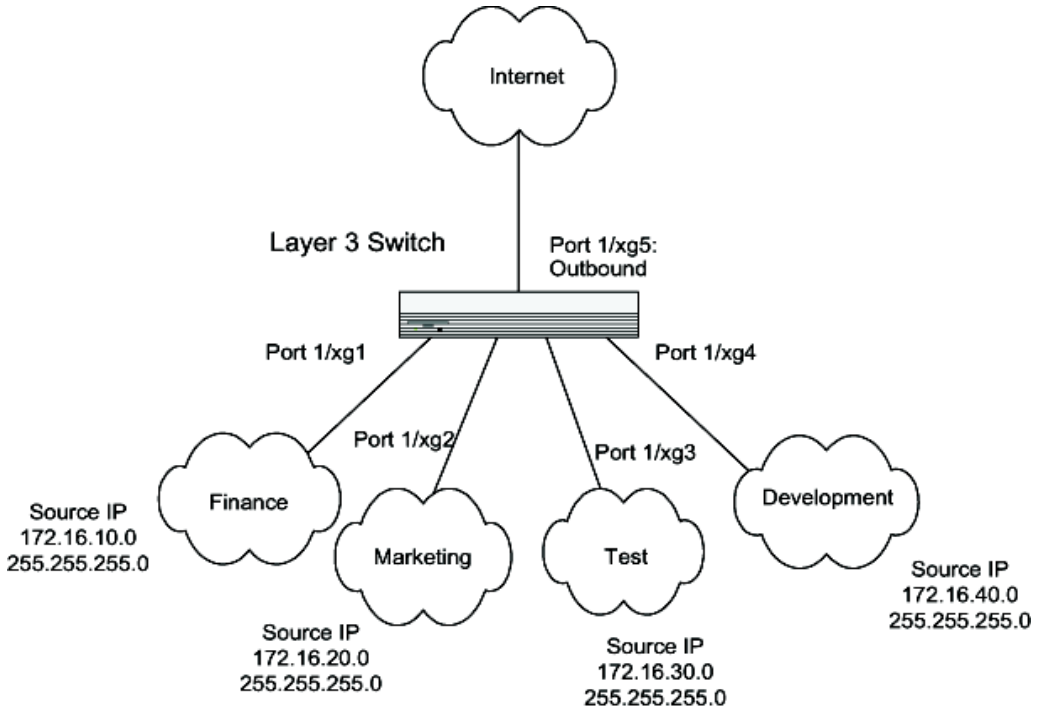


Figure 7-3. DiffServ Internet Access Example Network Diagram

Example #1: DiffServ Inbound Configuration

Ensure DiffServ operation is enabled for the switch.

```
console#config
  diffserv
```

Create a DiffServ class of type "all" for each of the departments, and name them. Define the match criteria—Source IP address—for the new classes.

```
   class-map match-all finance_dept
      match srcip 172.16.10.0 255.255.255.0
      exit

   class-map match-all marketing_dept
```

```
    match srcip 172.16.20.0 255.255.255.0
    exit

class-map match-all test_dept
    match srcip 172.16.30.0 255.255.255.0
    exit

class-map match-all development_dept
    match srcip 172.16.40.0 255.255.255.0
    exit
```

Create a DiffServ policy for inbound traffic named internet_access, adding the previously created department classes as instances within this policy. This policy uses the assign-queue attribute to put each department's traffic on a different egress queue. This is how the DiffServ inbound policy connects to the CoS queue settings established below.

```
policy-map internet_access in
    class finance_dept
        assign-queue 1
        exit
    class marketing_dept
        assign-queue 2
        exit
    class test_dept
        assign-queue 3
        exit
    class development_dept
        assign-queue 4
        exit
    exit
```

Attach the defined policy to interfaces 1/g1 through 1/g4 in the inbound direction

```
interface ethernet 1/g1
    service-policy in internet_access
    exit
interface ethernet 1/g2
    service-policy in internet_access
    exit
interface ethernet 1/g3
    service-policy in internet_access
    exit
interface ethernet 1/g4
    service-policy in internet_access
    exit
```

Set the CoS queue configuration for the (presumed) egress interface 1/g5 such that each of queues 1, 2, 3 and 4 get a minimum guaranteed bandwidth of 25%. All queues for this interface use weighted round robin scheduling by default. The DiffServ inbound policy designates that these queues are to be used for the departmental traffic through the assign-queue attribute. It is presumed that the switch will forward this traffic to interface 1/g5 based on a normal destination address lookup for internet traffic.

```
interface ethernet 1/g5
   cos-queue min-bandwidth 0 25 25 25 25 0 0
   exit
exit
```

DiffServ for VoIP Configuration Example

One of the most valuable uses of DiffServ is to support Voice over IP (VoIP). VoIP traffic is inherently time-sensitive: for a network to provide acceptable service, a guaranteed transmission rate is vital. This example shows one way to provide the necessary quality of service: how to set up a class for UDP traffic, have that traffic marked on the inbound side, and then expedite the traffic on the outbound side. The configuration script is for Router 1 in the accompanying diagram: a similar script should be applied to Router 2.

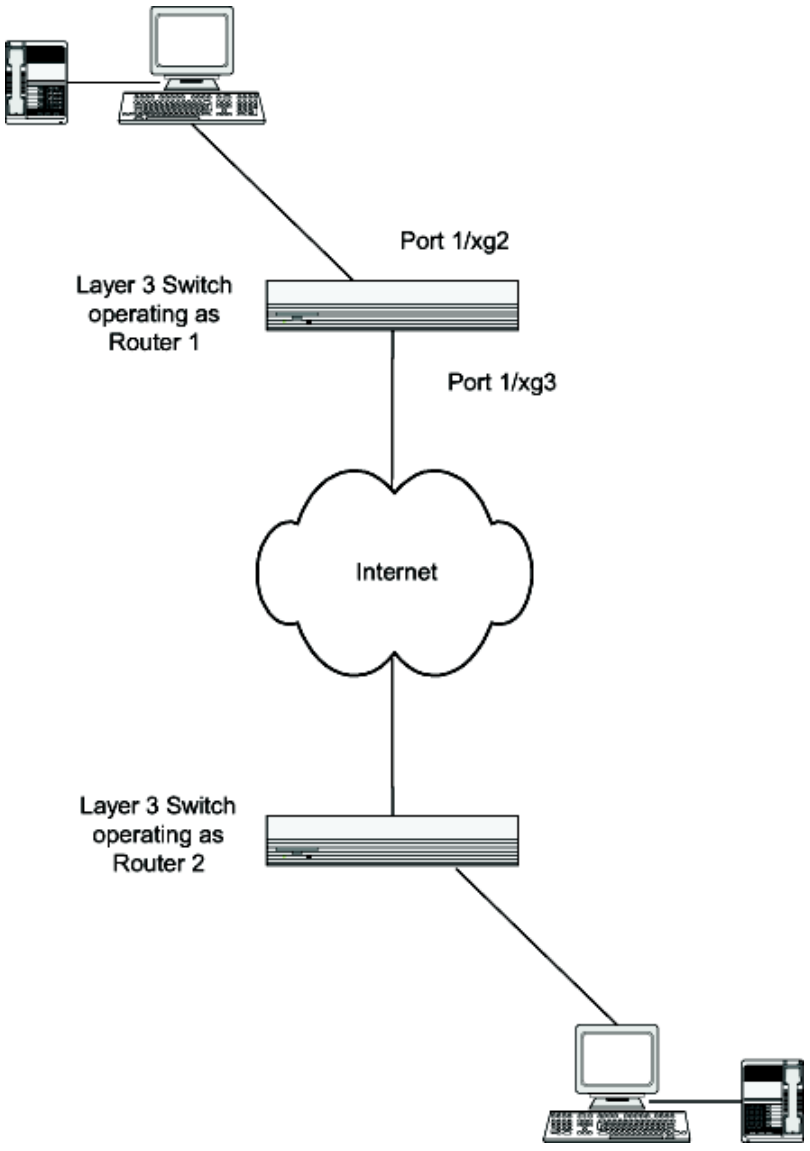


Figure 7-4. DiffServ VoIP Example Network Diagram

Example #2: Configuring DiffServ VoIP Support

Enter Global Config mode. Set queue 6 on all ports to use strict priority mode. This queue shall be used for all VoIP packets. Activate DiffServ for the switch.

```
console#config
   cos-queue strict 6
   diffserv
```

Create a DiffServ classifier named class_voip and define a single match criterion to detect UDP packets. The class type match-all indicates that all match criteria defined for the class must be satisfied in order for a packet to be considered a match.

```
   class-map match-all class_voip
      match protocol udp
      exit
```

Create a second DiffServ classifier named class_ef and define a single match criterion to detect a DiffServ code point (DSCP) of EF (expedited forwarding). This handles incoming traffic that was previously marked as expedited elsewhere in the network.

```
   class-map match-all class_ef
      match ip dscp ef
      exit
```

Create a DiffServ policy for inbound traffic named pol_voip, then add the previously created classes 'class_ef' and 'class_voip' as instances within this policy.

This policy handles incoming packets already marked with a DSCP value of EF (per class_ef definition), or marks UDP packets per the class_voip definition) with a DSCP value of EF. In each case, the matching packets are assigned internally to use queue 6 of the egress port to which they are forwarded.

```
   policy-map pol_voip in
      class class_ef
      assign-queue 5
      exit
   class class_voip
      mark ip-dscp ef
      assign-queue 5
      exit
   exit
```

Attach the defined policy to an inbound service interface.

```
interface ethernet 1/g1
service-policy in pol_voip
      exit
   exit
```

8

Multicast

This section provides configuration scenarios for the following features:

- "IGMP Configuration" on page 150
- "IGMP Proxy" on page 151
- "DVMRP" on page 152
- "PIM" on page 154
- "Multicast Routing and IGMP Snooping" on page 157

Overview

IP Multicasting enables a network host (or multiple hosts) to send an IP datagram to multiple destinations simultaneously. The initiating host sends each multicast datagram only once to a destination multicast group address, and multicast routers forward the datagram only to hosts who are members of the multicast group. Multicast enables efficient use of network bandwidth, as each multicast datagram needs to be transmitted only once on each network link, regardless of the number of destination hosts. Multicasting contrasts with IP unicasting, which sends a separate datagram to each recipient host.

Hosts must have a way to identify their interest in joining any particular multicast group, and routers must have a way to collect and maintain group memberships: these functions are handled by the IGMP protocol in IPv4. In IPv6, multicast routers use the Multicast Listener Discover (MLD) protocol to maintain group membership information.

Multicast routers must also be able to construct a multicast distribution tree that enables forwarding multicast datagrams only on the links that are required to reach a destination group member. Protocols such as DVMRP, and PIM handle this function.

IGMP is a multicast group discovery protocol that is used between the clients and the local multicast router. PIM-SM, PIM-DM, and DVMRP are multicast routing protocols that are used across different subnets, usually between the local multicast router and remote multicast router.

When to Enable IP Multicast on the PowerConnect 6200 Series Switch

Use the IP multicast feature on the PowerConnect 6200 Series switch to route multicast traffic between VLANs on the switch. If all hosts connected to the switch are on the same subnet, there is no need to configure the IP multicast feature. If the switch does not handle L3 routing, you can use IGMP snooping to manage port-based multicast group membership. For more information, see "IGMP Snooping" on page 40. If the local network does not have a multicast router, you can configure the switch to act as the IGMP querier. For more information, see "IGMP Snooping Querier" on page 43

IGMP Configuration

The Internet Group Management Protocol (IGMP) is used by IPv4 hosts to send requests to join (or leave) multicast groups so that they receive (or discontinue receiving) packets sent to those groups.

In IPv4 multicast networks, multicast routers are configured with IGMP so that they can receive join and leave request from directly-connected hosts. They use this information to build a multicast forwarding table.

IPv6 multicast routers use the MLD protocol to perform the functions that IGMP performs in IPv4 networks.

CLI Example

The following example configures IGMP on a PowerConnect 6200 Series switch in order to route multicast traffic between VLANs. IP routing, IP multicasting, and IGMP are globally enabled on the router. Then, IGMP is configured on the selected interface(s).

```
console#configure
    ip routing
    ip multicast
    ip igmp
    interface vlan 2
        routing
        ip address 3.3.3.1 255.255.255.0
        ip igmp
        exit
    exit
```

A multicast router must also have a way to determine how to efficiently forward multicast packets. The information gathered by IGMP is provided to a multicast routing protocol (i.e., DVMRP, PIM-DM, and PIM-SM) configured on the router to ensure that multicast packets are delivered to all networks where there are interested receivers. Refer to those sections for configuration instructions.

IGMP Proxy

IGMP proxy enables a multicast router to learn multicast group membership information and forward multicast packets based upon the group membership information. The IGMP Proxy is capable of functioning only in certain topologies that do not require Multicast Routing Protocols (i.e., DVMRP, PIM-DM, and PIM-SM) and have a tree-like topology, as there is no support for features like reverse path forwarding (RPF) to correct packet route loops.

The proxy contains many downstream interfaces and a unique upstream interface explicitly configured. It performs the host side of the IGMP protocol on its upstream interface and the router side of the IGMP protocol on its downstream interfaces.

The IGMP proxy offers a mechanism for multicast forwarding based only on IGMP membership information. The router must decide about forwarding packets on each of its interfaces based on the IGMP membership information. The proxy creates the forwarding entries based on the membership information and adds it to the multicast forwarding cache (MFC) in order not to make the forwarding decision for subsequent multicast packets with same combination of source and group.

CLI Examples

The CLI component of the Dell switch allows the end users to configure the network device and to view device settings and statistics using a serial interface or telnet session.

Example #1: Configuring IGMP Proxy on the Router

This command enables the IGMP Proxy on the router. To enable IGMP Proxy on the router no multicast routing protocol should be enabled and also multicast forwarding must be enabled on the router. Use these commands from the Interface mode:

```
console#configure
    ip routing
    ip multicast
    ip igmp
    interface vlan 15
       ip igmp-proxy
```

Additional configuration options are available for the igmp-proxy command:

```
<cr>                          Press Enter to execute the command.
reset-status                  Reset All the proxy interface status
                              parameters.
unsolicited-report-interval   Configure IGMP Proxy unsolicited report
                              interval.
```

The value of the unsolicited report interval can range from 1 to 260 seconds. The default is 1 second. Use this command from the Interface mode.

Example #2: View IGMP Proxy Configuration Data

You can use various commands from Privileged EXEC or User EXEC modes to show IGMP proxy configuration data.

- Use the following command to display a summary of the host interface status parameters. It displays the parameters only when IGMP Proxy is enabled.

```
console#show ip igmp-proxy
Interface Index................................ vlan 15
Admin Mode..................................... Enabled
Operational Mode............................... Disabled
```

- Use the following command to display interface parameters when IGMP Proxy is enabled:

```
console#show ip igmp-proxy interface
```

- Use this command to display information about multicast groups that IGMP proxy reported. It displays a table of entries with the following as the fields of each column.

```
console#show ip igmp-proxy groups
```

- Use the following command to display information about multicast groups that IGMP proxy reported. It displays a table of entries with the following as the fields of each column:

```
console#show ip igmp-proxy groups detail
```

DVMRP

The Distance Vector Multicast Routing Protocol (DVMRP) is one of several multicast routing protocols you can configure on the switch (PIM-SM and PIM-DM are the others). Note that only one multicast routing protocol (MRP) can be operational on a router at any time.

DVMRP is an interior gateway protocol; i.e., it is suitable for use within an autonomous system, but not between different autonomous systems.

DVMRP is based on RIP: it forwards multicast datagrams to other routers in the AS and constructs a forwarding table based on information it learns in response. More specifically, it uses this sequence.

- A new multicast packet is forwarded to the entire multicast network, with respect to the time-to-live (TTL) of the packet.
- The TTL restricts the area to be flooded by the message.
- All routers that do not have members on directly-attached subnetworks send back *Prune messages* to the upstream router.
- The branches that transmit a prune message are deleted from the delivery tree.
- The delivery tree which is spanning to all the members in the multicast group, is constructed in the form of a DVMRP forwarding table.

CLI Example

The following example configures two DVMRP interfaces. First, this example configures an OSPF router[1] and globally enables IP routing and IP multicast. IGMP is globally enabled so that this router can manage group membership information for its directly-connected hosts (IGMP may not be required when there are no directly connected hosts). Next, DVMRP is globally enabled. Finally, DVMRP, IGMP, and OSPF are enabled on several interfaces.

```
console#configure
   router ospf
      router-id 3.3.1.1
      exit
   ip routing
   ip multicast
   ip igmp
   ip dvmrp
   interface vlan 15
      routing
      ip address 3.3.3.1 255.255.255.0
      ip dvmrp
      ip igmp
      ip ospf area 0
      exit
   interface vlan 30
      routing
      ip address 1.1.1.1 255.255.255.0
      ip dvmrp
      ip igmp
      ip ospf area 0
      exit
   exit
```

1. OSPF configuration is added as a unicast protocol for illustration purposes; static unicast routing could also be configured.

PIM

Protocol Independent Multicast (PIM) is a standard multicast routing protocol that provides scalable inter-domain multicast routing across the Internet, independent of the mechanisms provided by any particular unicast routing protocol.

PIM has two types:

- PIM-Dense Mode (PIM-DM)
- PIM-Sparse Mode (PIM-SM)

PIM-SM

PIM-SM is used to efficiently route multicast traffic to multicast groups that may span wide area networks where bandwidth is a constraint.

PIM-SM uses shared trees by default and implements source-based trees for efficiency; it assumes that no hosts want the multicast traffic unless they specifically ask for it. It creates a shared distribution tree centered on a defined rendezvous point (RP) from which source traffic is relayed to the receivers. Senders first send the multicast data to the RP, which in turn sends the data down the shared tree to the receivers. Shared trees centered on an RP do not necessarily provide the shortest, most optimal path. In such cases, PIM-SM provides a means to switch to more efficient source-specific trees. A data threshold rate is configured to determine when to switch from shared-tree to source-tree.

PIM-SM uses a Bootstrap Router (BSR), which advertises information to other multicast routers about the RP. In a given network, a set of routers can be administratively enabled as candidate bootstrap routers. If it is not apparent which router should be the BSR, the candidates flood the domain with advertisements. The router with the highest priority is elected. If all the priorities are equal, then the candidate with the highest IP address becomes the BSR.

PIM-SM is defined in RFC 4601.

Example: PIM-SM

The following example configures PIM-SM for IPv4 on a router.

First, configure an OSPF[1] router and globally enable IP routing, multicast, IGMP, and PIM-SM. Next, configure a PIM-SM rendezvous point with an IP address and group range. The IP address will serve as an RP for the range of potential multicast groups specified in the group range. Finally, enable routing, IGMP, PIM-SM, and OSPF on one or more interfaces.

```
console#configure
   router ospf
      router-id 3.3.1.1
      exit
   ip routing
   ip multicast
   ip igmp
   ip pimsm            [NOTE: This router should be an RP.]
   ip pimsm rp-address 1.1.1.1 224.0.0.0 240.0.0.0
   interface vlan 15
      routing
      ip address 3.3.3.1 255.255.255.0
      ip pimsm
      ip igmp
      ip ospf area 0
      exit
   interface vlan 30
      routing
      ip address 1.1.1.1 255.255.255.0
      ip pimsm
      ip igmp
      ip ospf area 0
      exit
   exit
```

PIM-DM

PIM-DM protocol is a simple, protocol-independent multicast routing protocol. It uses existing unicast routing table and join/prune/graft mechanism to build a tree. PIM-DM creates source-based shortest-path distribution trees making use of Reverse Path Forwarding (RPF).

PIM-DM cannot be used to build a shared distribution tree, as PIM-SM can. PIM-DM assumes that when a sender starts sending data, all downstream routers and hosts want to receive a multicast datagram. PIM-DM initially floods multicast traffic throughout the network. Routers that do not have any downstream neighbors send back Prune messages that instruct the upstream router to remove that multicast route from its forwarding table. In addition to the Prune messages, PIM-DM makes use of two more messages: Graft and Assert. Graft messages are used whenever a new host wants to join the group. Assert messages are used to shut off duplicate flows onto the same multi-access network.

1. OSPF configuration is added as a unicast protocol for illustration purposes; static unicast routing could also be configured.

To minimize the repeated flooding of datagrams and subsequent pruning associated with a particular source-group (S,G) pair, PIM-DM uses a State Refresh message. This message is sent by the router(s) directly connected to the source and is propagated throughout the network. When received by a router on its RPF interface, the State Refresh message causes an existing prune state to be refreshed. State Refresh messages are generated periodically by the router directly attached to the source.

PIM-DM is appropriate for:

- Densely distributed receivers
- A ratio of few senders-to-many receivers (due to frequent flooding)
- High volume of multicast traffic
- Constant stream of traffic

Example: PIM-DM

The following example configures PIM-DM for IPv4 on a router.

First, configure an OSPF[1] router and globally enable IP routing, multicast, IGMP, and PIM-DM. Next, enable routing, IGMP, PIM-DM, and OSPF on one more interfaces.

```
console#configure
   router ospf
      router-id 3.3.1.1
      exit
   ip routing
   ip multicast
   ip igmp
   ip pimdm
   interface vlan 1
      routing
      ip address 3.3.3.1 255.255.255.0
      ip pimdm
      ip igmp
      ip ospf area 0
      exit
   interface vlan 3
      routing
      ip address 1.1.1.1 255.255.255.0
      ip pimdm
      ip igmp
      ip ospf area 0
      exit
   exit
```

1. OSPF configuration is added as a unicast protocol for illustration purposes; static unicast routing could also be configured.

Multicast Routing and IGMP Snooping

In this example, ports 1/g5 and 1/g10 are members of VLAN 100, and port 1/g15 is a member of VLAN 200. Both VLANs are configured as VLAN routing interfaces and are in different subnets. IGMP snooping is configured on VLAN 100 so that a member port will receive multicast data only if it sends an IMGP join message for that multicast group. IGMP and PIM-DM are enabled on each VLAN so that multicast data sent from a port on VLAN 200 can be routed to VLAN 100.

NOTE: If multicast routing and IGMP snooping are configured on the same switch, do not use the commands **bridge multicast filtering forward-all** or **bridge multicast filtering forbidden** on the VLAN interfaces. These commands should be used in IGMP Snooping configurations only.

1 Create VLANs 100 and 200.

```
console#configure
console(config)#vlan database
console(config-vlan)#vlan  100,200
```

2 Enable IGMP snooping on VLAN 100.

```
console(config-vlan)#ip igmp snooping 100
console(config-vlan)#exit
```

3 Enable routing on the switch.

```
console(config)#ip routing
```

4 Configure VLAN 100 as a VLAN routing interface and assign an IP address and subnet mask.

```
console(config)#interface vlan 100
console(config-if-vlan100)#routing
console(config-if-vlan100)#ip address  10.10.10.1  255.255.255.0
```

5 Enable IGMP and PIM-DM on the VLAN routing interface

```
console(config-if-vlan100)#ip igmp
console(config-if-vlan100)#ip pimdm
```

NOTE: When you enable PIM-DM on the VLAN routing interface, PIM-SM is automatically enabled. This is a known limitation. Only one multicast routing protocol can be enabled globally on the switch at a time, so the PIM mode that is enabled globally is the PIM mode that is active on the interface.

6 Configure VLAN 200 as a VLAN routing interface and assign an IP address and subnet mask.

```
console(config)#interface vlan 200
console(config-if-vlan200)#routing
console(config-if-vlan200)#ip address  20.20.20.1  255.255.255.0
```

7 Enable IGMP and PIM-DM on the VLAN routing interface

```
console(config-if-vlan200)#ip igmp
console(config-if-vlan200)#ip pimdm
```

8 Globally enable IGMP snooping, IP multicast, IGMP, and PIM-DM on the switch.

```
console(config)#ip igmp snooping
console(config)#ip multicast
console(config)#ip igmp
console(config)#ip pimdm
```

NOTE: Only one multicast routing protocol (PIM-SM, PIM-DM, or DVMRP) can be enabled globally on the switch at a time.

9 Configure ports 1/g5 and 1/g10 as members of VLAN 100.

```
console(config)#interface ethernet 1/g5
console(config-if-1/g5)#switchport access vlan 100
console(config-if-1/g5)#exit
console(config)#interface ethernet 1/g10
console(config-if-1/g10)#switchport access vlan 100
console(config-if-1/g10)#exit
```

10 Configure ports 1/g15 as a member of VLAN 200.

```
console(config)#interface ethernet 1/g15
config-if-1/g15)#switchport access vlan 200
config-if-1/g15)#exit
console(config)#exit
```

The following commands show multicast and routing information before any IGMP joins or multicast data is sent. The commands are in bold text.

```
console#show bridge multicast address-table
There are currently no entries in the table.

console#show ip route
Route Codes: R - RIP Derived, O - OSPF Derived, C - Connected, S - Static
      B - BGP Derived, IA - OSPF Inter Area
      E1 - OSPF External Type 1, E2 - OSPF External Type 2
      N1 - OSPF NSSA External Type 1, N2 - OSPF NSSA External Type 2

C      10.10.10.0/24 [0/1] directly connected,   vlan 100
C      20.20.20.0/24 [0/1] directly connected,   vlan 200

console#show ip pimdm
Admin Mode..................................... Enabled

    PIM-DM INTERFACE STATUS

Interface  Interface-Mode  Operational-Status
---------  --------------  ----------------
vlan 100   Enabled         Operational
vlan 200   Enabled         Operational
```

```
console#show ip igmp
IGMP Admin Mode.............................. Enabled
IGMP Router-Alert check...................... Disabled

      IGMP INTERFACE STATUS
Interface  Interface-Mode  Operational-Status
---------  --------------  ----------------
vlan 100   Enabled          Operational
vlan 200   Enabled          Operational
```

The host connected to interface 1/g5 sends an IGMP join message for multicast group 225.1.1.1 in VLAN 100. Then, the host connected to 1/g15 sends multicast data for group 225.1.1.1 in VLAN 200. Interface 1/g5 sends and receives the multicast data from the host on interface 1/g15. The multicast traffic must be routed because the hosts on 1/g5 and 1/g15 are in different subnets. Due to IGMP snooping, interface 1/g10 does not send or receive any multicast data even though it is in the same broadcast domain as interface 1/g5.

```
console#show bridge multicast address-table

Vlan        MAC Address         Type      Ports
----  ----------------------  ------- ------------------
100   0100.5E01.0101           Dynamic    1/g5

      Forbidden ports for multicast addresses:
Vlan        MAC Address             Ports
----  ---------------------- ----------------------------
100   0100.5E01.0101

console#show ip mcast mroute summary

      Multicast Route Table Summary
                                          Incoming  Outgoing
Source IP       Group IP        Protocol  Interface Interface List
--------------- --------------- --------- --------- ---------------
20.20.20.20     225.1.1.1       PIMDM     vlan 200  vlan 100
```

9

Utility

This section describes the following features:

- "Auto Config" on page 162
- "Nonstop Forwarding on a Switch Stack" on page 168

Auto Config

Overview

Auto Config is a software feature that automatically configures a switch when the device is initialized and no configuration file is found on the switch. Auto Config is accomplished in three phases:

1. Assignment (configuration) of an IP address for the device
2. Assignment of a TFTP server
3. Obtaining a configuration file for the device from the TFTP server

Functional Description

The Auto Config feature initiates when a switch is turned on and the startup-config file is not found. Auto Config is successful when a configuration file is downloaded to the switch from a TFTP server.

The Auto Config process requires DHCP to be enabled by default.

NOTE: The downloaded configuration file is not automatically saved to startup-config. An administrator must explicitly issue a save request in order to save the configuration.

The Auto Config process depends upon the configuration of other devices in the network, including:

- DHCP or BOOTP server
- TFTP server
- DNS server (if necessary)

IP Address Assignment

If BOOTP or DHCP is enabled on the switch and an IP address has not been assigned, the switch issues requests for an IP address assignment. The behavior of BOOTP or DHCP with respect to IP address assignment is unchanged by the addition of the Auto Config feature. That is, the following information returned from the server is recognized:

- IP address (yiaddr) and subnet mask (option 1) to be assigned to the switch
- IP address of a default gateway (option 3), if needed for IP communication

After an IP address is assigned to the switch, if a hostname is not already assigned, Auto Config issues a DNS request for the corresponding hostname. This hostname is also displayed as the CLI prompt (as in response to the **hostname** command).

Assignment of TFTP Server

The following information is also processed, and may be returned by a BOOTP or DHCP server:

- Name of configuration file (bootfile or option 67) available for download from the TFTP server.
- Identification of the TFTP server providing the bootfile. This can be obtained from any of the following fields:

– The hostname of the TFTP server (option 66 or sname). Either the TFTP address or name is specified (not both) in most network configurations. If a TFTP hostname is given, a DNS server is required to translate the name to an IP address.
– The IP address of the TFTP server (option 150).
– The address of the TFTP server (siaddr) to be used for Auto Config requests.

No configuration assigned by BOOTP or DHCP is saved in startup-config.

A DNS server is needed to resolve the IP address of the TFTP server only if the sname or option 66 values are used.

Obtaining a Config File

After obtaining IP addresses for both the switch and the TFTP server, the Auto Config process attempts to download a configuration file. When possible, a host-specific configuration file is downloaded. Otherwise, a network configuration file is used to get the final configuration. The process is described below.

The switch attempts to download a host-specific configuration file if a bootfile name was specified by the DHCP or BOOTP server. The switch makes three unicast TFTP requests for the specified bootfile. If the unicast attempts fail, or if a TFTP server address was not provided, the switch makes three broadcast requests to any available TFTP server for the specified bootfile. A TFTP broadcast request is a simple TFTP request with broadcast destination MAC address (ff:ff:ff:ff:ff:ff) and destination IP address (255.255.255.255).

NOTE: The bootfile is required to have a file type of *.cfg.

Attempts are made to download a default network configuration file with the name "fp-net.cfg" when:

- the host-specific bootfile cannot be found.
- a failure occurs in the host-specific configuration file download.
- the switch was not provided a specific bootfile name by the DHCP server.

The switch unicasts or broadcasts TFTP requests for a network configuration file in the same manner as the attempts to download a host-specific configuration file.

The default network configuration file should have IP address to hostname mappings using the command **ip host <hostname> <address>**. If the default network configuration file does not contain the switch's IP address, the switch uses DNS to attempt to resolve its hostname.

A sample fp-net.cfg file follows:

```
config
  ...
  ip host switch_to_setup 192.168.1.10
  ip host another_switch 192.168.1.11
  ... <other hostname definitions>
exit
```

Once a hostname has been determined, the switch then issues a TFTP request for a file named "<hostname>.cfg" file, where <hostname> is the first 32 characters of the switch's hostname.

If the switch is unable to map its IP address to a hostname, Auto Config sends TFTP requests for the default configuration file "host.cfg."

Table 9-1 summarizes the config files which may be downloaded, and the order in which they are sought.

Table 9-1. Configuration File Possibilities

Order Sought	File Name	Description	Final File Sought
1	<bootfile>.cfg	Host-specific config file, ending in a *.cfg file extension	Yes
2	fp-net.cfg	Default network config file	No
3	<hostname>.cfg	Host-specific config file, associated with hostname	Yes
4	host.cfg	Default config file	Yes

Table 9-2 displays the determining factors for issuing unicast or broadcast TFTP requests.

Table 9-2. TFTP Request Types

TFTP Server Address Available	Host-specific Router Config Filename Available	TFTP Request Method
Yes	Yes	Issue a unicast request for the host-specific router config file to the TFTP server
Yes	No	Issue a unicast request for a default network or router config file to the TFTP server
No	Yes	Issue a broadcast request for the host-specific router config file to any available TFTP server
No	No	Issue a broadcast request for the default network or router config file to any available TFTP server

Monitoring and Completing the Auto Config Process

When a switch begins bootup and there is no saved configuration, a message appears on the console informing the user that the Auto Config procedure is starting. A message also appears when Auto Config completes. The user is reminded with a message indicating that configuration must be saved in order to avoid performing Auto Config on the next reboot.

When Auto Config has successfully completed, an administrator can execute a **show running-config** command to validate the contents of configuration.

Saving a Configuration

An administrator must explicitly save the downloaded configuration in non-volatile memory. This makes the configuration available for the next reboot. In the CLI, this is performed by issuing **copy running-config startup-config** command and should be done after validating the contents of saved configuration.

Host-Specific Config File Not Found

If the Auto Config process fails to download a configuration file, a message is logged. If a final configuration file is not downloaded, as described in Table 9-1, the Auto Config procedure continues to issue TFTP broadcast requests. The frequency of the broadcasts is once per 10 minute period.

Terminating the Auto Config Process

A user can terminate the Auto Config process at any time prior to the downloading of the config file. This is useful when the switch is disconnected from the network, or when the requisite configuration files are configured on TFTP servers. Termination of the Auto Config process ends further periodic requests for a host-specific file.

Managing Downloaded Config Files

The configuration files downloaded by Auto Config are stored in the nonvolatile memory. The files may be managed (viewed, displayed, deleted) along with files downloaded by the configuration scripting utility.

A file is not automatically deleted after it is downloaded. The file does not take effect upon a reboot unless an administrator opts to save config (the saved configuration takes effect upon reboot). If the user does not opt to save config, the Auto Config process occurs again on a subsequent reboot. This may result in one of the previously downloaded files being overwritten.

Restarting the Auto Config Process

The Auto Config process is automatically started on a subsequent reboot if the configuration file is not found on the switch. This can occur if configuration has not been saved on the switch, or after the administrator issues a command to erase the configuration file.

During a session, the Auto Config process may be restarted (if the administrator has previously stopped the Auto Config process). This action re-initiates the process for this login session only. It is recommended that this action be performed only when the administrator is certain that configuration is clear in order to have predictable results.

Reinitialization of the switch (after a clear config) automatically activates the Auto Config process if there is no configuration file stored on the switch.

Switch Configuration Considerations

BOOTP or DHCP must be enabled on the switch in order for the Auto Config procedure to operate.

Network Configuration Considerations

Specifying a Default Router

Some network configurations require the specification of a default gateway through which some IP communication can occur. The default gateway is specified by Option 3 of a BOOTP or DHCP response.

Dependency Upon Other Network Services

The Auto Config process depends upon the following network services:

- A DHCP or BOOTP server must be configured on the network with appropriate services.
- A configuration file for the switch must be available from a TFTP server on the network.
- The switch must be connected to the network.
- A DNS server must contain an IP address to hostname mapping for the TFTP server if the DHCP server response contains only the hostname for the TFTP server.
- A DNS server must contain an IP address to hostname mapping for the switch if a <hostname>.cfg file is to be downloaded.
- If a default gateway is needed to forward TFTP requests, IP helper addresses will need to be configured on the gateway to provide those services.

Other Functions

CLI Scripting

CLI scripting can apply config files. It can be used to manage (view, validate, delete) downloaded config files, query Auto Config status, and to stop or restart the feature.

Logging

A message is logged for each of the following events:

- Auto Config component receiving a config file name and other options upon resolving an IP address by DHCP or BOOTP client. The boot options values are logged.
- Auto Config component initiating a TFTP request for a boot (config) file, receiving the file, or timing out of that request. Filenames and server IP addresses/hostnames are logged.
- Auto Config component initiating a request for a hostname. IP addresses and resolved hostnames are logged.
- Auto Config component initiating a TFTP request for a "<hostname>.cfg" file, receiving the file, or timing out of that request. Filenames, server IP addresses, and hostnames are logged.
- Applying a config script.
- Failure of the CLI scripting utility to apply a config file.

SIM

The SIM stores the hostname of the switch. After the DNS client resolves the hostname, it configures the SIM with the hostname. The Auto Config component queries the SIM to obtain the hostname. The hostname is used for "<hostname>.cfg" file request from TFTP. This hostname is also displayed as the CLI prompt.

TFTP Client

The TFTP client downloads configuration files and sends TFTP requests to the broadcast IP address (255.255.255.255).

DNS Client

The DNS client resolves an IP address to a hostname and resolves a hostname to an IP address (reverse IP address to hostname mapping).

BOOTP/DHCP Client

The DHCP and BOOTP clients handle predefined IP address configuration. The DHCPINFORM message type is sent to request Auto Config boot options.

Stacking

The downloaded configuration file is not distributed across a stack. When an administrator saves configuration, the config file is distributed across a stack.

CLI Examples

Example 1: Show Auto Config Process

To display the current status of the Auto Config process, use the following command:

```
console#show boot

Config Download
via DHCP: enabled
Auto Config State : Waiting for boot options
...
Auto Config State : Resolving switch hostname
...
Auto Config State : Downloading file <boot>.cfg
```

Example 2: Enable Auto Config

To start or stop Auto Config on the switch, use the following commands:

console#boot host dhcp

console#no boot host dhcp

Nonstop Forwarding on a Switch Stack

Networking devices, such as the PowerConnect 6200 Series switches, are often described in terms of three semi-independent functions called the forwarding plane, the control plane, and the management plane. The forwarding plane forwards data packets and is implemented in hardware. The control plane is the set of protocols that determine how the forwarding plane should forward packets, deciding which data packets are allowed to be forwarded and where they should go. Application software on the management unit acts as the control plane. The management plane is application software running on the management unit that provides interfaces allowing a network administrator to configure the device.

The Nonstop Forwarding (NSF) feature allows the forwarding plane of stack units to continue to forward packets while the control and management planes restart as a result of a power failure, hardware failure, or software fault on the stack management unit. This type of operation is called nonstop forwarding. When the management unit fails, only the switch ASICs on the management unit need to be restarted.

To prevent adjacent networking devices from rerouting traffic around the restarting device, the NSF feature uses the following three techniques:

1. A protocol can distribute a part of its control plane to stack units so that the protocol can give the appearance that it is still functional during the restart.
2. A protocol may enlist the cooperation of its neighbors through a technique known as graceful restart.
3. A protocol may simply restart after the failover if neighbors react slowly enough that they will not normally detect the outage.

Initiating a Failover

The NSF feature allows you to initiate a failover, which results in a warm restart of the master unit in the stack. Initiating a failover reloads the management unit, triggering the backup unit to take over. Before the failover, the management unit pushes application data and other important information to the backup unit. Although the handoff is controlled and causes minimal network disruption, some application state is lost, such as pending timers and other pending internal events.

Checkpointing

Switch applications (features) that build up a list of data such as neighbors or clients can significantly improve their restart behavior by remembering this data across a warm restart. This data can either be stored persistently, as DHCP server and DHCP snooping store their bindings database, or the management unit can checkpoint this data directly to the backup unit. Persistent storage allows an application on a standalone unit to retain its data across a restart, but since the amount of storage is limited, persistent storage is not always practical.

The NSF checkpoint service allows the management unit to communicate certain data to the backup unit in the stack. When the stack selects a backup unit, the checkpoint service notifies applications to start a complete checkpoint. After the initial checkpoint is done, applications checkpoint changes to their data.

NOTE: The switch cannot guarantee that a backup unit has exactly the same data that the management unit has when it fails. For example, the management unit might fail before the checkpoint service gets data to the backup if an event occurs shortly before a failover.

Table 9-3 lists the applications on the switch that checkpoint data and describes the type of data that is checkpointed.

Table 9-3. Applications that Checkpoint Data

Application	Checkpointed Data
ARP	Dynamic ARP entries
Auto VOIP	Calls in progress
Captive Portal	Authenticated clients
DHCP server	Address bindings (persistent)
DHCP snooping	DHCP bindings database
DOT1Q	Internal VLAN assignments
DOT1S	Spanning tree port roles, port states, root bridge, etc.
DOT1X	Authenticated clients
DOT3ad	Port states
IGMP/MLD Snooping	Multicast groups, list of router ports, last query data for each VLAN
IPv6 NDP	Neighbor cache entries
iSCSI	Connections
LLDP	List of interfaces with MED devices attached
OSPFv2	Neighbors and designated routers
OSPFv3	Neighbors and designated routers
Route Table Manager	IPv4 and IPv6 dynamic routes
SIM	The system's MAC addresses. System up time. IP address, network mask, default gateway on each management interface, DHCPv6 acquired IPv6 address.
Voice VLAN	VoIP phones identified by CDP or DHCP (not LLDP)

Switch Stack MAC Addressing and Stack Design Considerations

The switch stack uses the MAC addresses[1] assigned to the management unit. If the backup unit assumes control due to a management unit failure or warm restart, the backup unit continues to use the original management unit's MAC addresses. This reduces the amount of disruption to the network because ARP and other L2 entries in neighbor tables remain valid after the failover to the backup unit.

Stack units should always be connected with a ring topology (or other biconnected topology), so that the loss of a single stack link does not divide the stack into multiple stacks. If a stack is partitioned such that some units lose all connectivity to other units, then both parts of the stack start using the same MAC addresses. This can cause severe problems in the network.

If you move the management unit of stack to a different place in the network, make sure you power down the whole stack before you redeploy the management unit so that the stack members do not continue to use the MAC address of the redeployed switch.

NSF Network Design Considerations

You can design your network to take maximum advantage of NSF. For example, by distributing a LAG's member ports across multiple units, the stack can quickly switch traffic from a port on a failed unit to a port on a surviving unit. When a unit fails, the forwarding plane of surviving units removes LAG members on the failed unit so that it only forwards traffic onto LAG members that remain up. If a LAG is left with no active members, the LAG goes down. To prevent a LAG from going down, configure LAGs with members on multiple units within the stack, when possible. If a stack unit fails, the system can continue to forward on the remaining members of the stack.

If your switch stack performs VLAN routing, another way to take advantage of NSF is to configure multiple "best paths" to the same destination on different stack members. If a unit fails, the forwarding plane removes Equal Cost Multipath (ECMP) next hops on the failed unit from all unicast forwarding table entries. If the cleanup leaves a route without any next hops, the route is deleted. The forwarding plane only selects ECMP next hops on surviving units. For this reason, try to distribute links providing ECMP paths across multiple stack units.

NSF Default Behavior

NSF is enabled by default. You can disable NSF in order to redirect the CPU resources consumed by data checkpointing. Checkpointing only occurs when a backup unit is elected, so there is no need to disable the NSF feature on a standalone switch. When a new unit is added to the stack, the new unit takes the configuration of the stack, including the NSF setting.

1. Each switch is assigned four consecutive MAC addresses. The system uses the first three MAC addresses for the service port, network port, and routing interfaces. The fourth MAC address is reserved for future use. A stack of switches uses the four MAC addresses.assigned to the management unit.

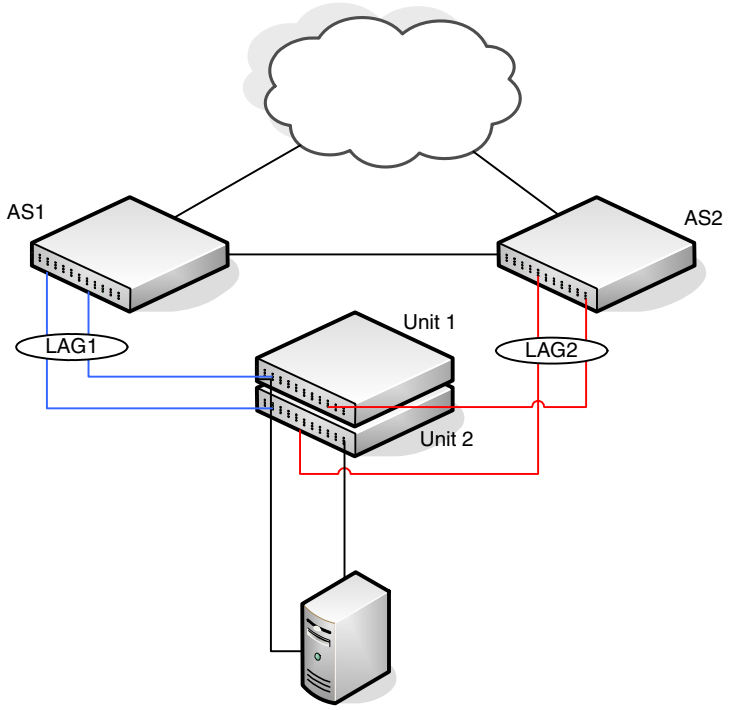
Configuration Examples

The actual configuration of the feature is simple. NSF is either enabled or disabled. The examples in this section describe how the NSF feature acts in various environments and with various switch applications.

Data Center

Figure 9-1 illustrates a data center scenario, where the stack of two PowerConnect 6200 Series switches acts as an access switch. The access switch is connected to two aggregation switches, AS1 and AS2. The stack has a link from two different units to each aggregation switch, with each pair of links grouped together in a LAG. The two LAGs and link between AS1 and AS2 are members of the same VLAN. Spanning tree is enabled on the VLAN. Assume spanning tree selects AS1 as the root bridge. Assume the LAG to AS1 is the root port on the stack and the LAG to AS2 is discarding. Unit 1 is the management unit. If unit 1 fails, the stack removes the unit 1 link to AS1 from its LAG. The stack forwards outgoing packets through the unit 2 link to AS1 during the failover. During the failover, the stack continues to send BPDUs and LAG PDUs on its links on unit 2. The LAGs stay up (with one remaining link in each), and spanning tree on the aggregation switches does not see a topology change.

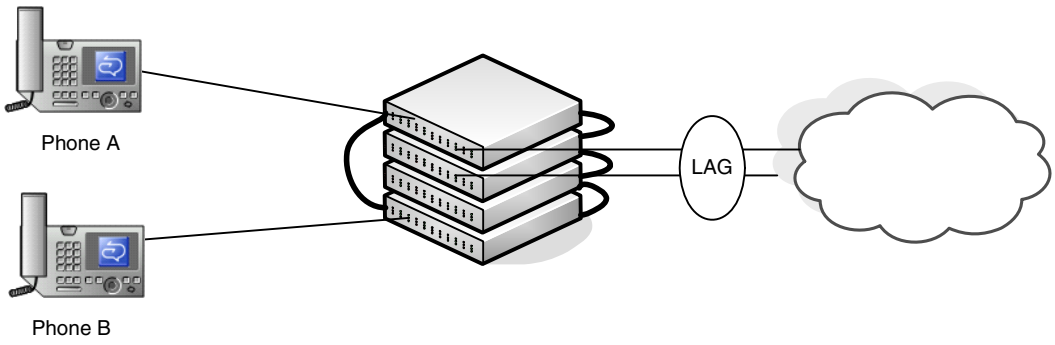
Figure 9-1. Data Center Stack Topology



VoIP

Figure 9-2 shows how nonstop forwarding maintains existing voice calls during a management unit failure. Assume the top unit is the management unit. When the management unit fails, the call from phone A is immediately disconnected. The call from phone B continues. On the uplink, the forwarding plane removes the failed LAG member and continues using the remaining LAG member. If phone B has learned VLAN or priority parameters through LLDP-MED, it continues to use those parameters. The stack resumes sending LLDPDUs with MED TLVs once the control plane restarts. Phone B may miss an LLDPDU from the stack, but should not miss enough PDUs to revert its VLAN or priority, assuming the administrator has not reduced the LLDPDU interval or hold count. If phone B is receiving quality of service from policies installed in the hardware, those policies are retained across the management unit restart.

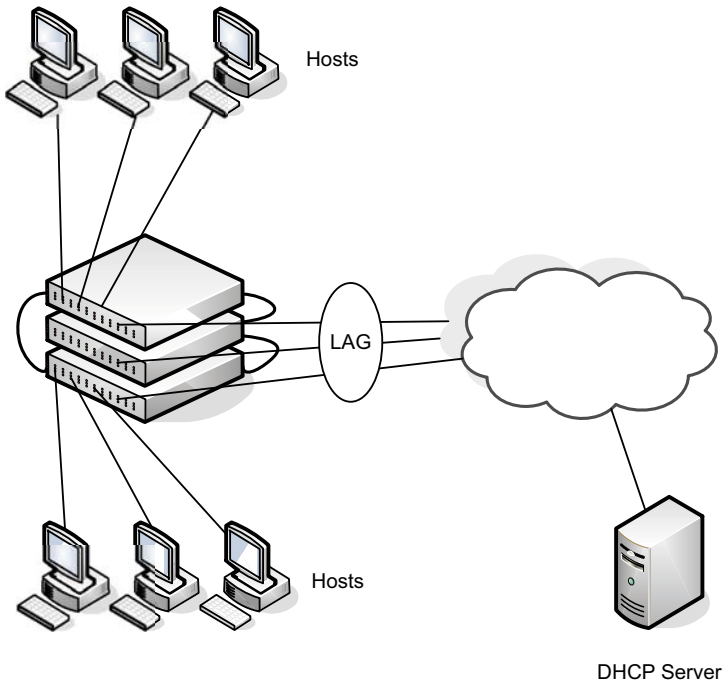
Figure 9-2. NSF and VoIP



DHCP Snooping Scenario

Figure 9-3 illustrates an L2 access switch running DHCP snooping. DHCP snooping only accepts DHCP server messages on ports configured as *trusted* ports. DHCP snooping listens to DHCP messages to build a bindings database that lists the IP address the DHCP server has assigned to each host. IP Source Guard (IPSG) uses the bindings database to filter data traffic in hardware based on source IP address and source MAC address. Dynamic ARP Inspection (DAI) uses the bindings database to verify that ARP messages contain a valid sender IP address and sender MAC address. DHCP snooping checkpoints its bindings database.

Figure 9-3. NSF and DHCP Snooping



If the management unit fails, all hosts connected to that unit lose network access until that unit reboots. The hardware on surviving units continues to enforce source filters IPSG installed prior to the failover. Valid hosts continue to communicate normally. During the failover, the hardware continues to drop data packets from unauthorized hosts so that security is not compromised.

If a host is in the middle of an exchange with the DHCP server when the failover occurs, the exchange is interrupted while the control plane restarts. When DHCP snooping is enabled, the hardware traps all DHCP packets to the CPU. The control plane drops these packets during the restart. The DHCP client and server retransmit their DHCP messages until the control plane has resumed operation and messages get through. Thus, DHCP snooping does not miss any new bindings during a failover.

As DHCP snooping applies its checkpointed DHCP bindings, IPSG confirms the existence of the bindings with the hardware by reinstalling its source IP address filters.

If Dynamic ARP Inspection is enabled on the access switch, the hardware traps ARP packets to the CPU on untrusted ports. During a restart, the control plane drops ARP packets. Thus, new traffic sessions may be briefly delayed until after the control plane restarts.

If IPSG is enabled and a DHCP binding is not checkpointed to the backup unit before the failover, that host will not be able to send data packets until it renews its IP address lease with the DHCP server.

Storage Access Network Scenario

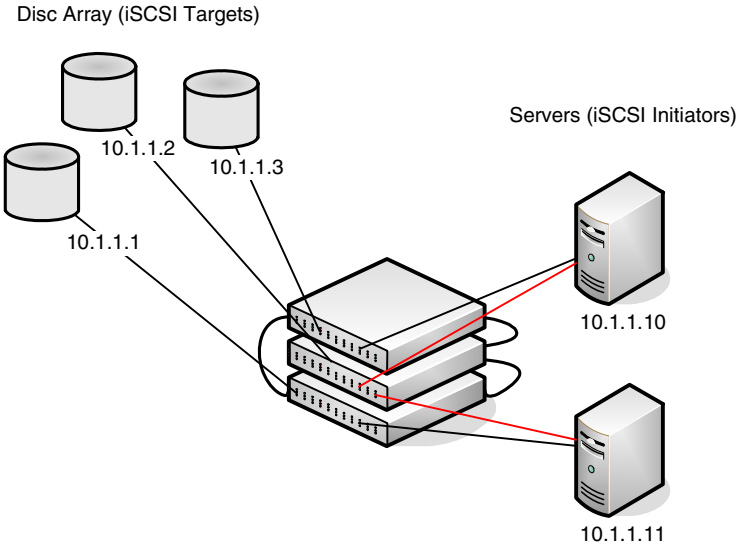
Figure 9-4 illustrates a stack of three PowerConnect 6200 Series switches connecting two servers (iSCSI initiators) to a disk array (iSCSI targets). There are two iSCSI connections as follows:

Session A: 10.1.1.10 to 10.1.1.3

Session B: 10.1.1.11 to 10.1.1.1

An iSCSI application running on the management unit (the top unit in the diagram) has installed priority filters to ensure that iSCSI traffic that is part of these two sessions receives priority treatment when forwarded in hardware.

Figure 9-4. NSF and a Storage Area Network



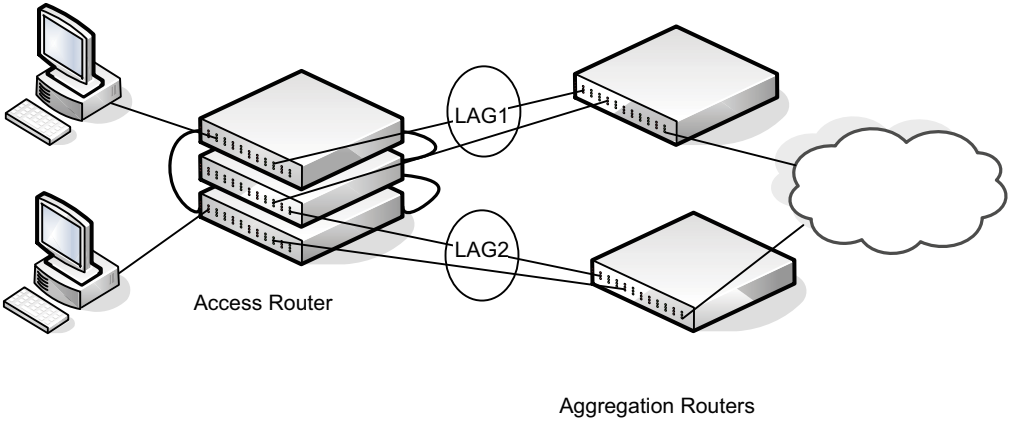
When the management unit fails, session A drops. The initiator at 10.1.1.10 detects a link down on its primary NIC and attempts to reestablish the session on its backup NIC to a different IP address on the disk array. The hardware forwards the packets to establish this new session, but assuming the session is established before the control plane is restarted on the backup unit, the new session receives no priority treatment in the hardware.

Session B remains established and fully functional throughout the restart and continues to receive priority treatment in the hardware.

Routed Access Scenario

Figure 9-5 shows a stack of three units serving as an access router for a set of hosts. Two LAGs connect the stack to two aggregation routers. Each LAG is a member of a VLAN routing interface. The stack has OSPF and PIM adjacencies with each of the aggregation routers. The top unit in the stack is the management unit.

Figure 9-5. NSF and Routed Access



If the management unit fails, its link to the aggregation router is removed from the LAG. When the control plane restarts, both routing interfaces come back up by virtue of the LAGs coming up. OSPF sends grace LSAs to inform its OSPF neighbors (the aggregation routers) that it is going through a graceful restart.

NOTE: The graceful restart feature for OSPF is disabled by default. To enable OSPF to perform a graceful restart for all planned and unplanned warm restart events, use the `nsf` command in Router OSPF Config mode:

```
console#configure
console(config)#router ospf
console(config-router)#nsf
```

The grace LSAs reach the neighbors before they drop their adjacencies with the access router. PIM starts sending hello messages to its neighbors on the aggregation routers using a new generation ID to prompt the neighbors to quickly resend multicast routing information. PIM neighbors recognize the new generation ID and immediately relay the group state back to the restarting router. IGMP sends queries to relearn the hosts' interest in multicast groups. IGMP tells PIM the group membership, and PIM sends JOIN messages upstream. The control plane updates the driver with checkpointed unicast routes. The forwarding plane reconciles L3 hardware tables.

The OSPF graceful restart finishes, and the control plane deletes any stale unicast routes not relearned at this point. The forwarding plane reconciles L3 multicast hardware tables. Throughout the process, the hosts continue to receive their multicast streams, possibly with a short interruption as the top aggregation router learns that one it its LAG members is down. The hosts see no more than a 50 ms interruption in unicast connectivity.